



UNIVERSIDAD DE LAS AMÉRICAS

FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

**INGENIERÍA EN CIBERSEGURIDAD E INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN**

ICBS0003 ANÁLISIS Y VISUALIZACIÓN DE DATOS

PROYECTO INTEGRADOR

Profesor:

Luis Patricio Moreno Buitrón

Integrantes:

María Alejandra Caicedo Mafla
Paul Sebastián Gutiérrez Muñoz

Fecha:

06 de julio del 2023

Tabla de Contenido

I.	Introducción	3
II.	Objetivos	3
III.	Descripción de la Solución Implementada.....	3
	• Implementación repositorios en una máquina virtual con Linux.....	3
	• Implementación en la última versión de MySQL.	3
	• Diseño físico	4
	• Diccionario de datos	4
	• Diseño dimensional de la bodega.....	5
	<input type="checkbox"/> Diseño lógico	5
	<input type="checkbox"/> Diseño físico	5
	<input type="checkbox"/> Diccionario de datos	5
	• Faker (https://faker.readthedocs.io/en/master/).....	6
	<input type="checkbox"/> Script	6
	<input type="checkbox"/> Tabla beneficiarios	7
	<input type="checkbox"/> Tabla empleados	7
	<input type="checkbox"/> Tabla proyectos	8
	• Los programas ETL deben ser implementados con Python.....	8
	• La capa de presentación de datos deberá ser implementada con Power BI	9
IV.	Instrucciones para acceder al repositorio en GitHub	10
V.	Informe de cumplimiento de los siguientes criterios de desempeño.....	11
VI.	Anexos	11

I. Introducción

Este documento presenta el proceso de desarrollo del proyecto, el cual se basa en la información recopilada del NRC (Consejo Noruego para Refugiados) con el objetivo de abordar las necesidades analíticas identificadas. Para lograr esto, se llevará a cabo la implementación de una base de datos y se han diseñado modelos de datos considerados óptimos para el proyecto. Como etapa final, se procederá a implementar la solución y se podrá acceder a ella visualmente mediante el uso de Power BI.

II. Objetivos

- Identificar las necesidades analíticas que se requieren solventar al interior de una organización
- Implementar la base de datos que será la fuente de datos para solventar las necesidades analíticas de la organización.
- Desarrollar el modelo de datos analítico en base a las necesidades identificadas previamente.
- Implementar una solución analítica para la adecuada y efectiva toma de decisiones al interior de una organización

III. Descripción de la Solución Implementada

- Implementación repositorios en una máquina virtual con Linux.

```
proyecto@proyecto-virtual-machine:~$ uname -r
5.19.0-46-generic
proyecto@proyecto-virtual-machine:~$ uname -a
Linux proyecto-virtual-machine 5.19.0-46-generic #47~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Wed Jun 21 15:35:31 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
proyecto@proyecto-virtual-machine:~$
```

- Implementación en la última versión de MySQL.

```
proyecto@proyecto-virtual-machine:~$ sudo mysql -u root -p NRC
[sudo] password for proyecto:
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

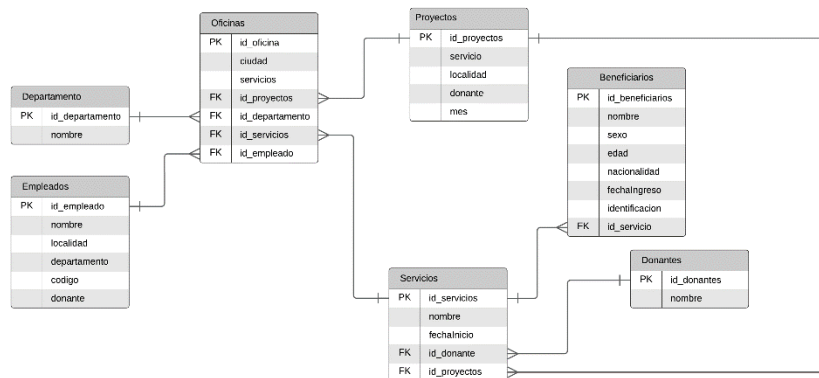
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 51
Server version: 8.0.33-0ubuntu0.22.04.2 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>
```

- Diseño físico

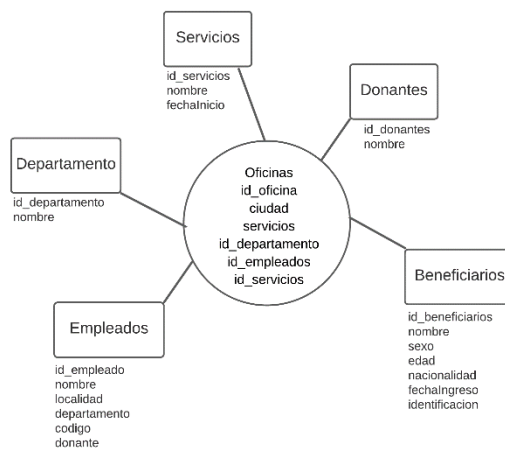


- Diccionario de datos

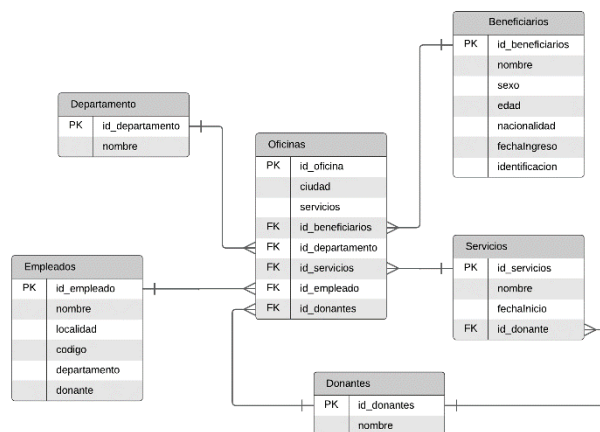
Tabla: Donantes				
	Dato	Tipo de dato	Descripción	
PK	id_donantes	INT	Identificador del donante	Los donantes validos son: Hilton, Acnur, FMSA,
	nombre	VARCHAR(50)	Nombre del donante	
Tabla: Servicios				
	Dato	Tipo de dato	Descripción	
PK	id_servicios	INT	Identificador del servicio	Los servicios serán Educación, FSL, ICLA, Vivienda, Asistencia
	nombre	VARCHAR(50)	Nombre del servicio	
	fechaInicio	DATE	Fecha en la que inicio el servicio	
FK	id_donante	INT	Identificador del donante	
FK	id_proyectos	INT	Identificador del proyecto	
Tabla: Beneficiarios				
	Dato	Tipo de dato	Descripción	
PK	id_beneficiario	INT	Identificador del beneficiario	
	nombre	VARCHAR(50)	Nombre del beneficiario	
	sexo	VARCHAR(10)	Genero del beneficiario	
	edad	CHAR(2)	Edad del beneficiario	
	nacionalidad	VARCHAR(50)	Nacionalidad del beneficiario	
	fechaIngreso	DATE	Fecha en la que ingreso el beneficiario	
	identificacion	CHAR(10)	Identificación perteneciente al beneficiario	
FK	id_servicio	INT	Identificador del servicio	
Tabla: Proyectos				
	Dato	Tipo de dato	Descripción	
PK	id_proyectos	INT	Identificador del proyecto	Las localidades podrán ser: Quito, San Lorenzo, Ibarra, Ambato, Guayaquil, Tulcan, Esmeraldas, Huaquillas, El coca, Cuenca, Manta, Sto
	servicio	VARCHAR(50)	Servicio correspondiente al proyecto	
	localidad	VARCHAR(50)	Localidad donde tendrá lugar el proyecto	
	donante	VARCHAR(10)	Donante parte del proyecto	
	mes	CHAR(2)	Meses del proyecto	
Tabla: Empleados				
	Dato	Tipo de dato	Descripción	
PK	id_empleado	INT	Identificador del empleado	Los departamentos validos son: RRHH, ICLA, MONITOREO, SOPORTE,
	nombre	VARCHAR(50)	Nombre del empleado	
	localidad	VARCHAR(50)	Localidad del empleado	
	departamento	VARCHAR(50)	Departamento al que pertenece el empleado	
	codigo	VARCHAR(10)	Codigo del empleado	
	donante	VARCHAR(10)	Donante del empleado	
Tabla: Oficinas				
	Dato	Tipo de dato	Descripción	
PK	id_oficina	INT	Identificador del empleado	
	ciudad	VARCHAR(50)	Ciudad donde se encuentra la oficina	
	servicios	VARCHAR(50)	Servicios que ofrece la oficina	
FK	id_proyectos	INT	Identificador del proyecto	
FK	id_departamento	INT	Identificador del departamento	
FK	id_empleados	INT	Identificador del empleado	
FK	id_servicios	INT	Identificador del servicio	
Tabla: Departamento				
	Dato	Tipo de dato	Descripción	
PK	id_departamento	INT	Identificador del departamento	
	nombre	VARCHAR(50)	Nombre del departamento	

- Diseño dimensional de la bodega

- Diseño lógico



- Diseño físico



- Diccionario de datos

Tabla: Donantes			
	Dato	Tipo de dato	Descripción
PK	id_donantes	INT	Identificador del donante
	nombre	VARCHAR(50)	Nombre del donante
Tabla: Servicios			
	Dato	Tipo de dato	Descripción
PK	id_servicios	INT	Identificador del servicio
	nombre	VARCHAR(50)	Nombre del servicio
	fechaInicio	DATE	Fecha en la que inicio el servicio
FK	id_donante	INT	Identificador del donante
FK	id_proyectos	INT	Identificador del proyecto
Tabla: Beneficiarios			
	Dato	Tipo de dato	Descripción
PK	id_beneficiarios	INT	Identificador del beneficiario
	nombre	VARCHAR(50)	Nombre del beneficiario
	sexo	VARCHAR(10)	Genero del beneficiario
	edad	CHAR(2)	Edad del beneficiario
	nacionalidad	VARCHAR(50)	Nacionalidad del beneficiario
	fechaIngreso	DATE	Fecha en la que ingreso el beneficiario
	identificacion	CHAR(10)	Identificación perteneciente al beneficiario
FK	id_servicio	INT	Identificador del servicio

Tabla: Empleados			
	Dato	Tipo de dato	Descripción
PK	id_empleado	INT	Identificador del empleado
	nombre	VARCHAR(50)	Nombre del empleado
	localidad	VARCHAR(50)	Localidad del empleado
	departamento	VARCHAR(50)	Departamento al que pertenece el empleado
	codigo	VARCHAR(10)	Código del empleado
	donante	VARCHAR(10)	Donante del empleado

Tabla: Oficinas			
	Dato	Tipo de dato	Descripción
PK	id_oficina	INT	Identificador del empleado
	ciudad	VARCHAR(50)	Ciudad donde se encuentra la oficina
	servicios	VARCHAR(50)	Servicios que ofrece la oficina
FK	id_proyectos	INT	Identificador del proyecto
FK	id_departament	INT	Identificador del departamento
FK	id_empleados	INT	Identificador del empleado
FK	id_servicios	INT	Identificador del servicio

Tabla: Departamento			
	Dato	Tipo de dato	Descripción
PK	id_departament	INT	Identificador del departamento
	nombre	VARCHAR(50)	Nombre del departamento

- Faker (<https://faker.readthedocs.io/en/master/>)
 - Script

```

GNU nano 6.2                                BDSetup.py *
import random
from faker import Faker
import mysql.connector
import datetime
from faker.providers import barcode

# Configuración de la conexión a la base de datos
config = {
    'user': 'root',
    'password': 'password',
    'host': 'localhost',
    'database': 'MRC',
    'raise_on_warnings': True
}

# Crear una conexión a la base de datos
conn = mysql.connector.connect(**config)
cursor = conn.cursor()

# Crear registros de ejemplo utilizando Faker y agregarlos a la tabla
fake = Faker()
fake.add_provider(barcode)

# CREACION TABLA DONANTES
cursor.execute("DROP TABLE IF EXISTS donantes")
cursor.execute("CREATE TABLE IF NOT EXISTS donantes (id_donantes INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(50), pais VARCHAR(50))")

def donantes():
    donante = ['Hilton', 'Acuru', 'FRSA', 'Innova', 'Gobierno']
    return donante

for _ in range(5):
    donante = donantes()
    nombre = donante
    pais = fake.country()
    cursor.execute("INSERT INTO donantes (nombre, pais) VALUES (%s, %s)", (nombre, pais))

# CREACION TABLA SERVICIOS
cursor.execute("CREATE TABLE servicios (id_servicios INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(50), fechaInicio DATE, id_donante INT, FOREIGN KEY (id_donante) REF")

# Definir una función para generar datos personalizados

```

```

GNU nano 6.2                                BDSetup.py *
# Definir una función para generar datos personalizados
def servicios():
    # Aquí puedes definir tu lógica para generar el dato personalizado
    servicio = ['Educación', 'FSL', 'ICLA', 'Vivienda', 'Asistencia']
    return servicio

for _ in range(5):
    servicio = servicios()
    nombre = servicio
    # Obtener la fecha actual
    fecha_actual = datetime.date.today()
    # Generar una fecha aleatoria en los últimos 5 años
    fechaInicio = fecha_actual - datetime.timedelta(days=365 * 5)
    cursor.execute("INSERT INTO servicios (nombre, fechaInicio) VALUES (%s, %s)", (nombre, fechaInicio))

# CREACION TABLA BENEFICIARIOS
cursor.execute("DROP TABLE IF EXISTS beneficiarios")
cursor.execute("CREATE TABLE beneficiarios (id_beneficiarios INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(50), sexo VARCHAR(10), edad CHAR(2), nacionalidad VARCHAR(50))")

for _ in range(50):
    random_number = random.randint(0, 1)
    if random_number == 0:
        # Si el número es 0, generar un nombre masculino
        nombre = fake.first_name_male()
        sexo = "Masculino"
    else:
        # Si el número es 1, generar un nombre femenino
        nombre = fake.first_name_female()
        sexo = "Femenino"
    # Generar una fecha de nacimiento aleatoria en los últimos 80 años
    fecha_nacimiento = fake.date_of_birth(minimum_age=1, maximum_age=80)
    # Calcular la edad en base a la fecha de nacimiento
    hoy = datetime.date.today()
    edad = hoy.year - fecha_nacimiento.year
    nacionalidad = fake.country()
    # Obtener la fecha actual
    fecha_actual = datetime.date.today()
    # Generar una fecha aleatoria en los últimos 20 años
    fecha_inicio = fecha_actual - datetime.timedelta(days=365 * 20)
    # Restar 20 años a la fecha actual
    fechaIngreso = fake.date_between(start_date=fecha_inicio, end_date=fecha_actual)

```

- Tabla beneficiarios

```
mysql> select * from beneficiarios;
```

id	nombre	sexo	edad	nacionalidad	fechaIngreso	identificacion
1	Scott	Masculino	47	Lesotho	2010-05-08	5914775768
2	Antonio	Masculino	51	BenIn	2012-11-26	6094130391
3	Diane	Femenino	41	Canada	2021-05-31	698600158
4	Judith	Femenino	57	Holy See (Vatican City State)	2009-05-11	9684451122
5	Steven	Masculino	70	Morocco	2019-06-02	6735876301
6	Steven	Masculino	32	Isle of Man	2004-02-10	8909838316
7	Miguel	Masculino	7	Samoa	2009-04-01	788500528
8	Steven	Masculino	12	Anguilla	2010-12-20	7645031640
9	Brian	Masculino	63	Slovakia (Slovak Republic)	2020-06-18	2246674849
10	Ryan	Masculino	24	Netherlands Antilles	2009-05-18	2959580966
11	Kendra	Femenino	56	Thailand	2004-03-27	3440711709
12	Samuel	Masculino	17	Guinea-Bissau	2005-05-16	7123838122
13	Thomas	Masculino	73	Gabon	2023-05-14	9761000325
14	Glen	Masculino	53	Albania	2018-05-25	634348324
15	Brittany	Femenino	60	Egypt	2014-09-13	4713186785
16	Ann	Femenino	65	Northern Mariana Islands	2005-08-11	8794190065
17	Rebecca	Femenino	42	Serbia	2010-07-05	1032390774
18	Julie	Femenino	17	France	2020-07-11	9585658985
19	Natasha	Femenino	75	Timor-Leste	2021-01-23	2291610967
20	Cody	Masculino	39	Saint Barthelemy	2013-07-29	4525476268
21	John	Masculino	2	Mozambique	2016-11-14	9145281939
22	Kaitlyn	Femenino	23	Bosnia and Herzegovina	2015-09-30	4728506791
23	Jennifer	Femenino	7	Liberia	2021-10-01	1349931691
24	Miguel	Masculino	63	Mongolia	2004-12-09	9261230234
25	Darrell	Masculino	42	British Virgin Islands	2017-09-18	9912491912
26	Willie	Masculino	9	Andorra	2018-03-10	9549929221
27	Maria	Femenino	77	Kiribati	2022-11-01	6927174886
28	Mary	Femenino	18	Guinea-Bissau	2013-02-01	3890178011
29	Brenda	Femenino	79	Qatar	2017-11-02	8703177629
30	Kayla	Femenino	22	Sri Lanka	2016-11-06	7173033674
31	Robert	Masculino	23	Egypt	2022-12-22	4132676002
32	Monica	Femenino	53	Bouvet Island (Bouvetoya)	2016-06-25	5388493839
33	Christopher	Masculino	12	American Samoa	2007-11-12	9791952734
34	Marcus	Masculino	24	Dominica	2004-05-05	3678586400
35	Tiffany	Femenino	9	France	2012-04-05	2007161460
36	Kyle	Masculino	24	Northern Mariana Islands	2021-05-25	2404723523
37	Douglas	Masculino	4	Eritrea	2016-11-08	9309615264
38	Whitney	Femenino	59	Luxembourg	2004-11-16	9066101761
39	Cristian	Masculino	26	Reunion	2011-05-30	7503180374
40	Jordan	Masculino	10	British Virgin Islands	2012-08-03	5061635711

- Tabla empleados

```
mysql> select * from empleados;
```

id	nombre	localidad	departamento	CODIGO	donante
1	Jack	Tulcan	RRHH	60247940	Innova
2	Jordan	Sto Domingo	ICLA	66082590	Innova
3	Harold	Esmeraldas	SOPORTE	71609584	Acnur
4	Tina	Ibarra	LOGISTICA	10937761	FMSA
5	Brian	Manta	RRHH	21920646	Innova
6	Taylor	Quito	ICLA	93188920	Hilton
7	James	San Lorenzo	EDUCACION	68055479	Acnur
8	Thomas	El coca	LOGISTICA	96765708	Innova
9	Mary	El coca	SOPORTE	95797021	Hilton
10	James	Esmeraldas	SOPORTE	11441571	FMSA
11	Kimberly	Ibarra	MONITOREO	34971178	Innova
12	Donald	Sto Domingo	RRHH	29796977	Acnur
13	Aaron	Esmeraldas	ICLA	80343493	Acnur
14	Charles	San Lorenzo	LOGISTICA	26644028	Acnur
15	Eric	Huaquillas	RRHH	18704365	Acnur
16	Michael	Sto Domingo	EDUCACION	13749552	Gobierno
17	Michael	Ibarra	SOPORTE	88068688	Innova
18	Michael	Quito	ICLA	65264959	Hilton
19	Alan	Manta	LOGISTICA	35010487	Innova
20	Tammy	Ambato	ICLA	39852588	Innova
21	John	El coca	MONITOREO	62616645	Innova
22	Crystal	Quito	RRHH	71012667	FMSA
23	Christopher	Sto Domingo	ICLA	89146958	Gobierno
24	Kenneth	Manta	ICLA	89741252	Innova
25	Darryl	Esmeraldas	LOGISTICA	99905248	Innova
26	Pamela	Manta	LOGISTICA	84296498	Hilton
27	John	El coca	MONITOREO	25919677	Hilton
28	Isabella	Ambato	ICLA	34302675	FMSA
29	Michelle	Cuenca	LOGISTICA	01128512	Innova
30	David	El coca	SOPORTE	92097438	Acnur
31	Timothy	Tulcan	RRHH	73236450	FMSA
32	Christopher	Guayaquil	SOPORTE	05434800	Acnur
33	Albert	Sto Domingo	SOPORTE	71606132	FMSA
34	Ashlee	Ibarra	LOGISTICA	23744646	Hilton
35	Gary	Sto Domingo	LOGISTICA	99405625	Innova
36	Alexis	Guayaquil	SOPORTE	16680227	Hilton
37	Kevin	Esmeraldas	EDUCACION	31124355	FMSA
38	Rhonda	Huaquillas	LOGISTICA	38379833	FMSA
39	Ian	El coca	RRHH	74201631	Gobierno
40	Ann	San Lorenzo	EDUCACION	97700357	Hilton

- Tabla proyectos

```
mysql> select * from proyectos;
```

id	servicio	localidad	donante	meses
1	ICLA	Huaquillas	Acnur	42
2	Asistencia	El coca	Gobierno	10
3	Vivienda	Manta	Innova	32
4	Vivienda	Ambato	FMSA	13
5	ICLA	Ibarra	Hilton	3
6	Asistencia	Huaquillas	Hilton	1
7	Educacion	San Lorenzo	FMSA	9
8	ICLA	San Lorenzo	Gobierno	44
9	Educacion	Huaquillas	FMSA	42
10	Vivienda	Guayaquil	Gobierno	40
11	Vivienda	Ibarra	Hilton	5
12	Vivienda	San Lorenzo	Acnur	26
13	Educacion	Esmeraldas	Acnur	9
14	Asistencia	Sto Domingo	Hilton	37
15	FSL	Cuenca	Innova	35
16	Asistencia	Tulcan	Acnur	21
17	Asistencia	Manta	Innova	34
18	Educacion	Huaquillas	Hilton	32
19	Vivienda	El coca	Hilton	5
20	ICLA	El coca	Innova	48
21	FSL	El coca	Acnur	47
22	Educacion	Cuenca	Innova	40
23	ICLA	Cuenca	Gobierno	46
24	Vivienda	Huaquillas	Hilton	13
25	Vivienda	Sto Domingo	Acnur	42
26	Asistencia	Huaquillas	Acnur	21
27	Educacion	Esmeraldas	Acnur	42
28	Asistencia	Huaquillas	Acnur	44
29	Asistencia	Cuenca	Acnur	42
30	FSL	Huaquillas	Innova	35
31	FSL	Tulcan	Hilton	19
32	Educacion	Ambato	Innova	19
33	Asistencia	Ambato	Acnur	9
34	ICLA	San Lorenzo	Hilton	41
35	Vivienda	Huaquillas	FMSA	18
36	FSL	Sto Domingo	Acnur	20
37	ICLA	Guayaquil	Acnur	32
38	FSL	Cuenca	Gobierno	32
39	Educacion	Huaquillas	Acnur	13
40	Asistencia	El coca	Acnur	16

- Los programas ETL deben ser implementados con Python

```
GNU nano 6.2 ext_refugiados.py
import traceback
from sqlalchemy import create_engine
import pandas as pd

def extraer_beneficiarios():
    try:
        host = 'localhost'
        port = '3306'
        user = 'root'
        pwd = 'password'
        db = 'NRC'

        # Crear el objeto de conexión utilizando SQLAlchemy
        conn_str = f'mysql+mysqlconnector://{user}:{pwd}@{host}:{port}/{db}'
        engine = create_engine(conn_str)

        # Leer los datos de la tabla 'country'
        beneficiarios = pd.read_sql('SELECT * FROM beneficiarios', con=engine)
        return beneficiarios

    except:
        traceback.print_exc()

    finally:
        pass
```



```

GNU nano 6.2 trans_donantes.py
import traceback
import pandas as pd
from datetime import datetime

def transformar_donantes (donantes):
    try:
        donantes['pais'] = ('EEUU')
        return donantes
    except:
        traceback.print_exc()
    finally:
        pass

```

```

GNU nano 6.2 carg_departamentos.py
import traceback
from util.db_conn import Db_connection
import psycopg2
import pandas as pd
from sqlalchemy import text
import traceback
from sqlalchemy import create_engine
import pandas as pd

def cargar_beneficiarios():
    try:
        host = 'localhost'
        port = '3306'
        user = 'root'
        pwd = 'password'
        db = 'HRC'

        # Crear el objeto de conexión utilizando SQLAlchemy
        conn_str = f'mysql+mysqlconnector://{user}:{pwd}@{host}:{port}/{db}'
        engine = create_engine(conn_str)

        # Leer los datos de la tabla ext_category en un dataframe
        data = pd.read_sql("SELECT id_beneficiarios, name FROM ext_beneficiarios", ses_db_stg)

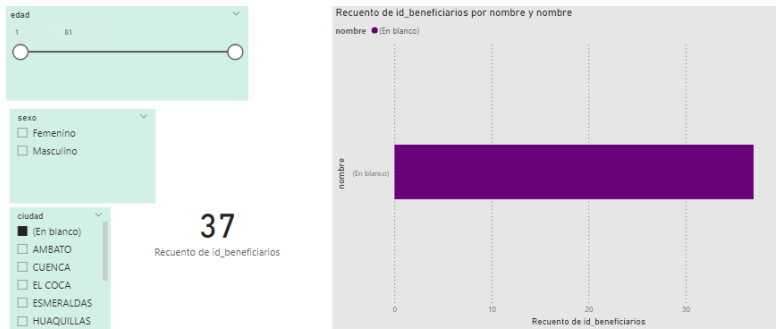
        # Renombrar las columnas del DataFrame para que coincidan con los nombres de columna de la tabla en la base de datos de destino
        data = data.rename(columns={"id_beneficiarios": "cat_bus_id", "name": "nombre"})

        # Crear TABLA temporal en sor_dvrental
        sql_temp = text("""
        CREATE TABLE IF NOT EXISTS temp_beneficiarios (
            CAT_BUS_ID INTEGER NOT NULL,
            NOMBRE VARCHAR(100) NOT NULL
        )
        """)
        with ses_db.sor.begin() as conn:
            conn.execute(sql_temp)

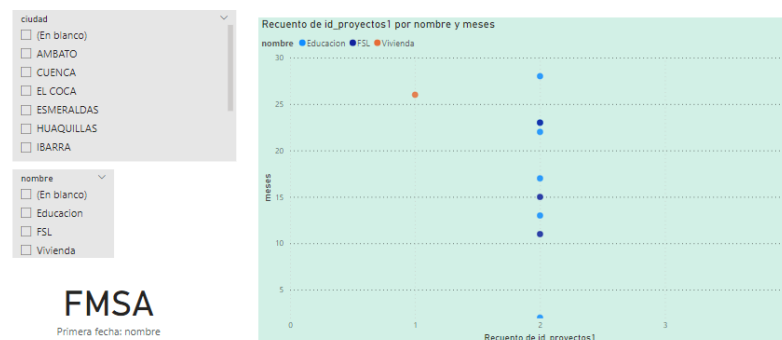
```

- La capa de presentación de datos deberá ser implementada con Power BI

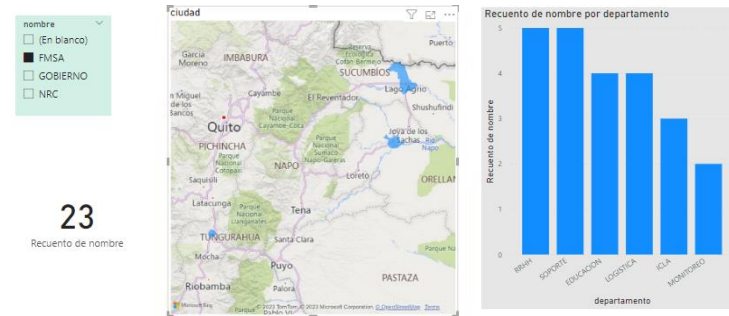
¿Cuántos Beneficiarios en base a su género y rango de edad acceden a cada uno de los servicios en las distintas localidades?



¿Qué Donante tiene mayor presencia en las localidades, con qué servicio y durante cuanto tiempo?



En base al departamento ¿Cuántos empleados trabajan bajo cada donante y donde están ubicados ?

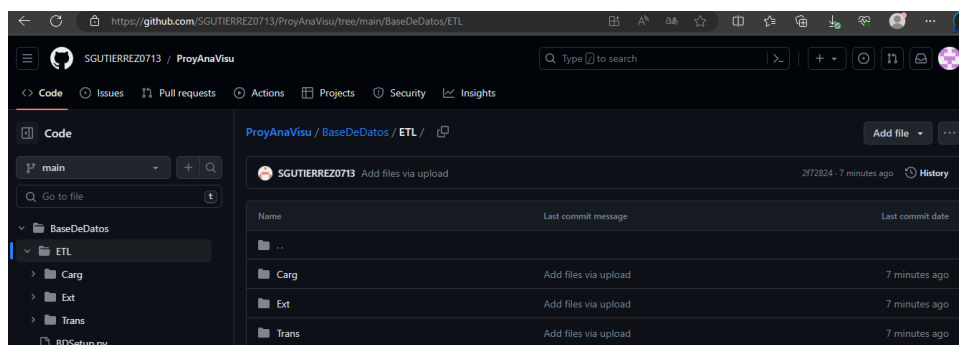
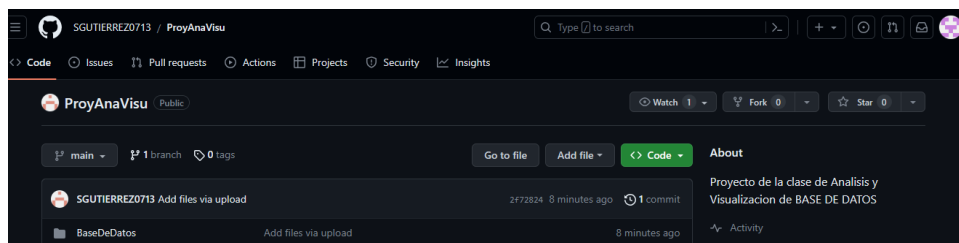


¿Qué servicio es el más utilizado en cada localidad?



IV. Instrucciones para acceder al repositorio en GitHub

Para ingresar al repositorio de GitHub lo que se debe hacer es abrir un navegador, e ingresar el siguiente link: <https://github.com/SGUTIERREZ0713/ProyAnaVisu>



V. Informe de cumplimiento de los siguientes criterios de desempeño

Criterio	Porcentaje de Cumplimiento	Evidencia(s)
Interpretación de la información visualizada	100%	Anexo 1: Interpretación de visualización
Obtención de conclusiones tras la implementación de la solución analítica	100%	Anexo 2: Conclusiones solución analítica

VI. Anexos

1. Interpretación de visualización

Análisis de tendencias: Al visualizar los datos a lo largo del tiempo, es posible identificar tendencias y patrones en la migración de refugiados. Esto puede ayudar a comprender si hay un aumento o disminución en el número de refugiados y si existen períodos o estacionalidades específicas.

Segmentación de datos demográficos: Al segmentar los datos demográficos de los refugiados, como la edad, el género o la nacionalidad, es posible obtener información detallada sobre grupos específicos. Esto puede revelar diferencias en las necesidades, desafíos o solicitudes de asistencia de cada grupo, lo que podría influir en las decisiones y estrategias de la organización.

Análisis geográfico: Al visualizar los datos en un mapa, es posible identificar áreas geográficas con una alta concentración de refugiados. Esto puede ayudar a la organización a enfocar sus esfuerzos y recursos en áreas específicas donde la necesidad es mayor.

Rendimiento de los programas de ayuda: Al monitorear los indicadores clave de desempeño, como la efectividad de los programas de ayuda, es posible visualizar métricas como la tasa de éxito en la integración de los refugiados, la satisfacción del cliente o la calidad de los servicios proporcionados. Esto puede permitir a la organización identificar áreas de mejora y tomar medidas para optimizar sus programas y servicios.

Análisis comparativo: Al comparar diferentes métricas o grupos de refugiados, es posible identificar relaciones y correlaciones interesantes. Por ejemplo, se puede comparar la edad de los refugiados con la tasa de empleo o el nivel educativo para evaluar si existen asociaciones significativas. Esto puede ayudar a la organización a comprender mejor las necesidades y desafíos de cada grupo y adaptar sus intervenciones en consecuencia.

2. Conclusiones solución analítica

Análisis de tendencias: La solución permitirá identificar tendencias y patrones en los datos, lo cual puede ayudar a comprender mejor la dinámica de los refugiados y proporcionar información relevante para la toma de decisiones.

Segmentación de datos: Mediante el análisis de datos demográficos, geográficos u otros, es posible segmentar la información y obtener una visión más detallada de los diferentes grupos de refugiados.

Seguimiento de indicadores clave: La solución analítica permitirá el monitoreo de indicadores clave de desempeño, como el número de refugiados atendidos, la eficacia de los programas de ayuda, entre otros.

Generación de informes automatizados: Power BI facilita la generación de informes automatizados