PRESTAR SERVICIOS PROFESIONALES PARA ATENDER TEMAS TECNICOS RELACIONADOS PARA LAELABORACIÓNDEDOCUMENTOSDELINEAMIENTOS TÉCNICOS, DE LOS PROGRAMAS, PROYECTOS, CONTRATOS Y/O CONVENIOS A CARGODELINVIAS

Configuración infraestructura: Sistema de procesamiento

INSTITUTO NACIONAL DE VÍAS TERRITORIAL ANTIOQUIA
PROYECTO: MAPA DE VULNERABILIDAD FAUNÍSTICA

Avance del segundo mes de trabajo desarrollo del entorno de trabajo y desarrollo, capacitación del equipo de trabajo en el uso del entorno de trabajo, integración de las funciones de cálculo y aplicación del modelo de procesamiento, arquitectura del sistema del proyecto
del Mapa de vulnerabilidad Faunística

Ing. Jairo I. Coy
Equipo de DevOps

CONFIGURACIÓN SISTEMA PROCESAMIENTO

El documento muestra las etapas de configuración del entorno de trabajo (sistema de procesamiento), como también las tecnologías para poder correrlo de forma apropiada.

El sistema de procesamiento requiere de los siguientes componentes para poder funcionar.

Miniconda: Ambiente virtual aislado, el cual permitirá almacenar todas los componentes tecnológicos y librerías que posteriormente serán instalados en el sistema de procesamiento.

Debido a las limitaciones y a la falta de compatibilidad con el sistema de gestión de paquetes del lenguaje R, el proyecto no se manejará con PIP.

R comprimido: Para poder correr los scripts en lenguaje R, se hace la compresión del motor de procesamiento de R, el cual se encargará de ejecutar el modelo MAXENT.

Para la instalación de paquetes y librerías requisitos del sistema, se debe ejecutar la instalación de todas librerías listadas en el archivo.

librerias_sistema_procesamiento

Adicional se crea la estructura y parámetros para establecer la conexión entre el sistema de procesamiento y el contenedor en la nube. Para esta etapa, el contenedor en la nube no es el oficial, pero presenta las mismas características que el oficial, por lo que al momento de realizar el cambio de contenedor no debe presentará mayores inconvenientes.

README

Pasos para correr el entorno de trabajo Django

- python3 -m venv venv
- pip install -r requirements.txt
- python -m manage.py runserver

Pasos para instalación de R en el server

- sudo apt update
- sudo apt install --no-install-recommends software-properties-common dirmngr -y
- wget -qO- https://cloud.r-project.org/bin/linux/ubuntu/marutter_pubkey.asc | sudo tee -a /etc/apt/trusted.gpg.d/cran_ubuntu_key.asc
- sudo add-apt-repository "deb https://cloud.r-project.org/bin/linux/ubuntu noble-cran40/"
- sudo apt update
- sudo apt install r-base r-base-dev -y

Paso con conda

- ir a https://docs.conda.io/en/latest/miniconda.html https://www.anaconda.com/download
- conda --version
- conda create -n r_env python=3.10 r-base rpy2 -c conda-forge
- conda activate r_env
- python script.py

Instalación de paquetes del proyecto con Conda

- conda install -c conda-forge r_env [package]
- conda activate r_env

Creación de archivo de requisitos

- pip freeze > requirements.txt
- conda list --export > requirements.txt

Permisos de acceso

- earthengine authenticate
- ~/.config/earthengine/credentials

CONECCIÓN CON EL CONTENEDOR EN LA NUBE

```
# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/5.2/howto/static-files/

STATIC_URL = 'static/'
# Development line
STATICFILES_DIRS = [os.path.join(BASE_DIR, 'static')]
# STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')

MEDIA_URL = '/media/'
MEDIA_ROOT = os.path.join(BASE_DIR, 'media')

# STATIC_ROOT = os.path.join(BASE_DIR, 'static')

# Production line
# STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')

# Url del lenguaje R
# RSCRIPT_PATH = r""

# Default primary key field type
# https://docs.djangoproject.com/en/5.2/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

# Google storage
DEFAULT_FILE_STORAGE = 'storages.backends.gcloud.GoogleCloudStorage'
GS_BUCKET_NAME = 'invias'
GS_PROJECT_ID = 'complete-energy-448804-i2'
```

1

2

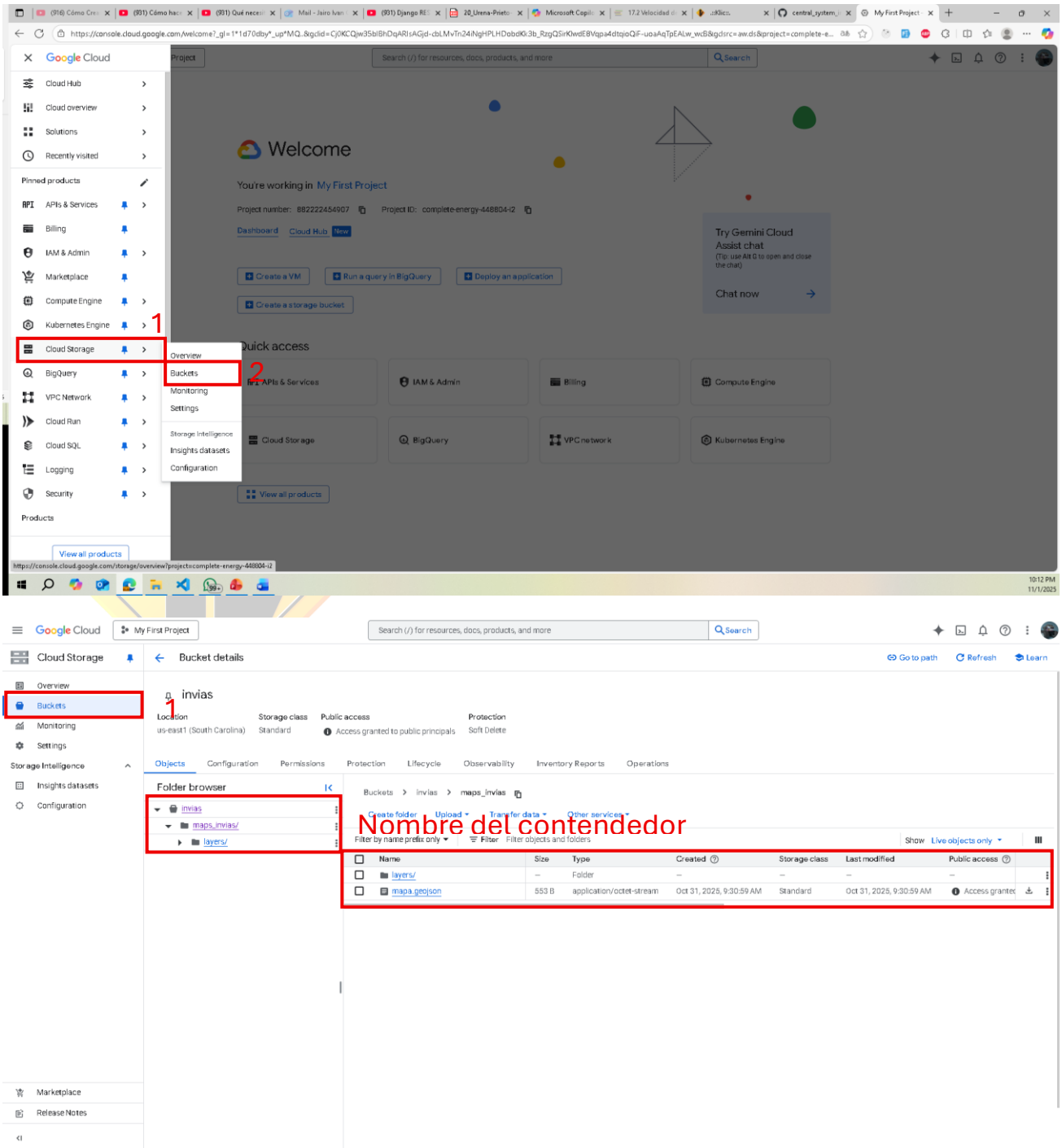
1: Las líneas encerradas, permiten que el modelo de procesamiento almacene la información en el mismo sistema.

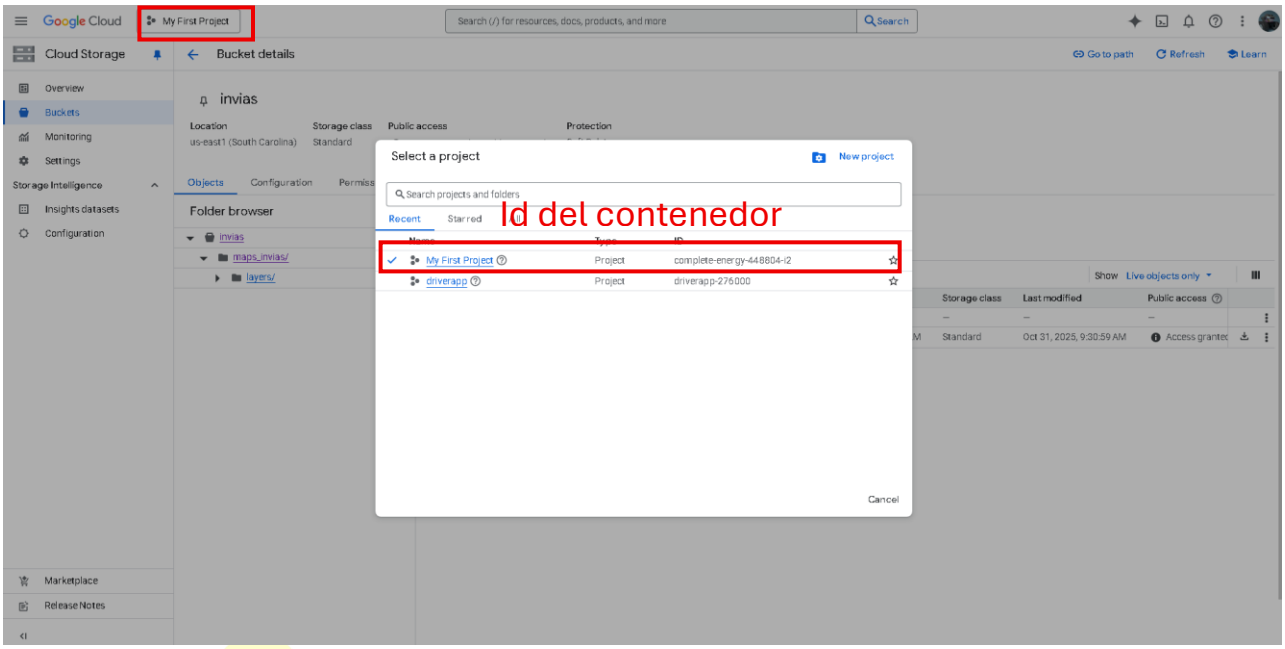
2: Las líneas encerradas, permiten conectar el contenedor en la nube, con el sistema de procesamiento.

La segunda línea se ingresa el nombre del contenedor. Y en la tercera el identificador.

La información, se adquiere directamente desde el contenedor, como se muestra en las imágenes.

Para ello es necesario entrar a la consola de Google Cloud Engine y buscar el proyecto y luego el contenedor.





En caso de presentarse un problema con el tráfico de datos cruzados, se debe verificar que la librería “django-cors-headers”. Esta librería brinda los permisos para poder establecer una conexión con sistemas externos y su respectivo tráfico de información entre sistemas.

Para configurar la librería de forma adecuada, posterior a la instalación en el archivo “settings.py” ingresar.

```python
ALLOWED_HOSTS = ['*']

CORS_ALLOW_ALL_ORIGINS = True

# CORS

CORS_ALLOWED_ORIGINS = [
    "http://localhost:3000",
    "http://127.0.0.1:9000",
]

# CORS ALLOW

CORS_ALLOW_METHODS = [
    'DELETE',
    'GET',
    'OPTIONS',
    'PATCH',
    'POST',
    'PUT',
]

CORS_ALLOW_HEADERS = [
    'accept',
    'authorization',
    'content-type',
    'user-agent',
    'x-csrftoken',
    'x-requested-with',
]
```

```python
# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'rest_framework',
    'corsheaders',
    'storages',
    'frontend',
    'backend',
    'demos',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
    'corsheaders.middleware.CorsMiddleware',
]
```