

Universidad Nacional de Misiones

**Licenciatura en Sistemas de Información
Analista en Sistemas de Computación
Profesorado Universitario en Computación**

ACTUALIDAD INFORMÁTICA

Equipo de Cátedra:

- Adjunto a Cargo: Mgter. Sergio Daniel Caballero
- JTP: Mgter. Martín Rey
- Ayudante 1º: Prof. ASC Alejandra Gonzales

Trabajo Práctico Nro. 5 - Gestión de entornos

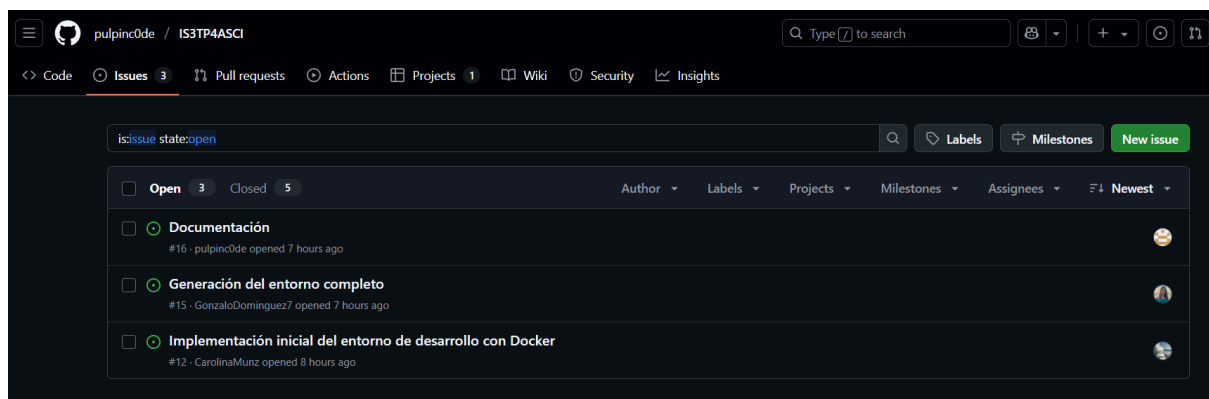
Integrantes:

- Dominguez Gonzalo,
- Münz Carolina,
- Pereyra Maribel

Consigna:

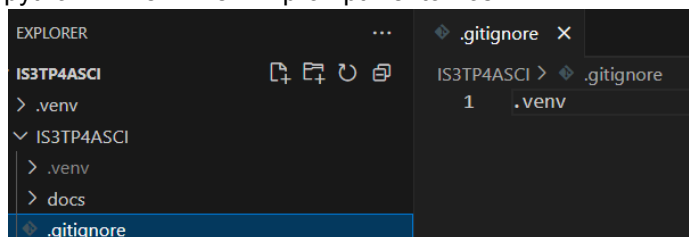
Replicar un ejemplo práctico del uso de Docker y Docker Compose para la gestión de entornos en un proyecto de desarrollo de software.

1. Sobre el mismo repositorio del TP4, generar un issue nuevo para hacer la implementación inicial del entorno de desarrollo. Asignarlo a uno de los integrantes del grupo. Hacer lo mismo con la generación del entorno completo (otro issue) y la documentación en sí del TP (otro issue más).



2. Hacer el clonado del repositorio a una copia local en un equipo donde se encuentre instalado Docker (opcionalmente se podrán usar GitHub Codespaces o alguna solución similar)

```
git clone https://github.com/pulpinc0de/IS3TP4ASCI.git
cd IS3TP4ASCI
python -m venv .venv --prompt="entornos"
```



3. Seleccionar un lenguaje y/o framework y buscar una imagen aplicable.

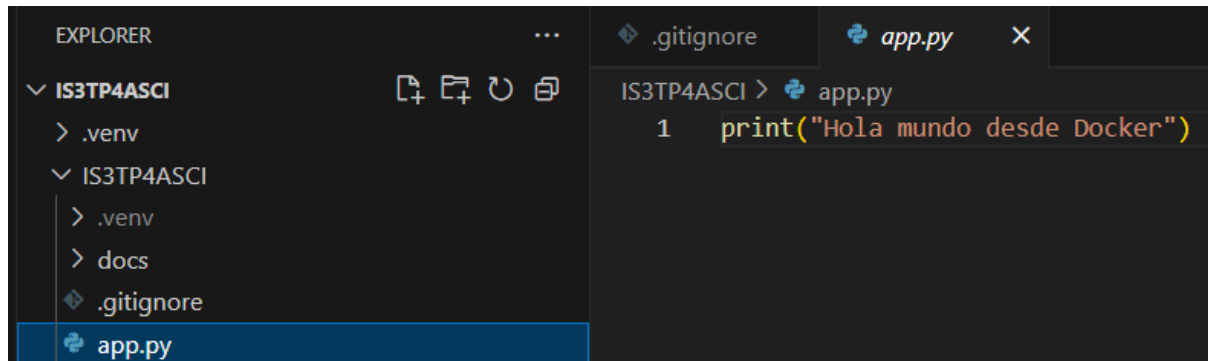
Usaremos Python 3.11. Docker tiene imágenes oficiales.

Stack definido:

- Python
- Base de Datos: PostgreSQL

4. Escribir un archivo similar a los usados en clase para realizar un “Hola mundo” en ese contexto.

Crear un archivo llamado app.py



The screenshot shows the VS Code interface. On the left, the Explorer sidebar displays the file structure of a project named 'IS3TP4ASCI'. It includes a '.venv' directory, a 'docs' directory, and files named '.gitignore' and 'app.py'. The 'app.py' file is selected and highlighted. On the right, the Editor pane shows the content of 'app.py', which is a single line of Python code: `print("Hola mundo desde Docker")`.

Crear el Dockerfile:



The screenshot shows the VS Code Editor with a file named 'dockerfile' open. The code is as follows:

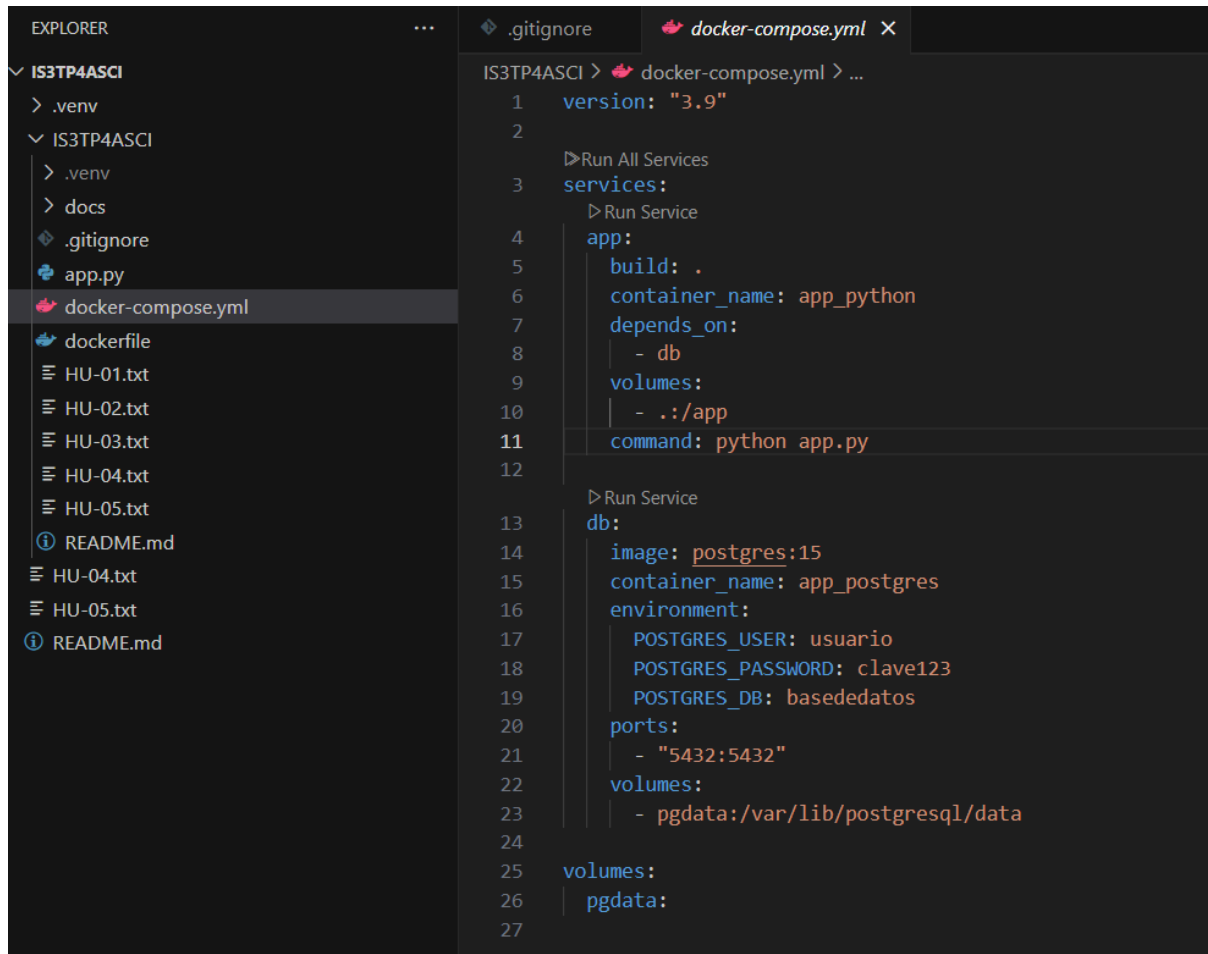
```
1  # Imagen base con Python
2  FROM python:3.11-slim
3
4  # Carpeta de trabajo dentro del contenedor
5  WORKDIR /app
6
7  # Copiamos el archivo Python
8  COPY app.py .
9
10 # Comando que se ejecuta al iniciar el contenedor
11 CMD ["python", "app.py"]
12
```

5. Probar su ejecución con un contenedor de la forma vista en la clase con un Dockerfile.

```
docker build -t tp5-python-app .
docker run tp5-python-app
```

6. Plantear otros servicios a ser agregados y conformar un archivo `docker-compose.yml` donde al menos se conecte un servicio para la aplicación y uno para persistencia (se puede seleccionar cualquier motor de base de datos).

Crear archivo `docker-compose.yml`:

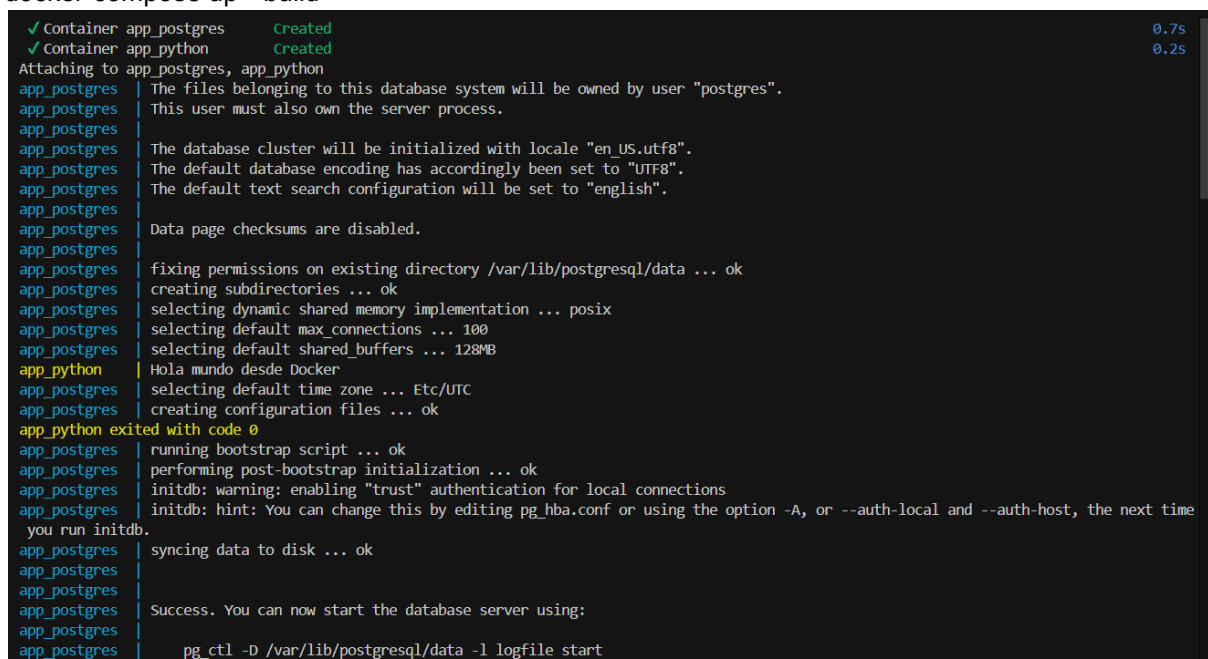


```

EXPLORER
  IS3TP4ASCI
    .venv
    IS3TP4ASCI
      .venv
      docs
      .gitignore
      app.py
      docker-compose.yml
      dockerfile
      HU-01.txt
      HU-02.txt
      HU-03.txt
      HU-04.txt
      HU-05.txt
      README.md
      HU-04.txt
      HU-05.txt
      README.md

IS3TP4ASCI > docker-compose.yml > ...
1  version: "3.9"
2
3  >Run All Services
4  services:
5    >Run Service
6    app:
7      build: .
8      container_name: app_python
9      depends_on:
10       - db
11     volumes:
12       - ../app
13     command: python app.py
14
15   >Run Service
16   db:
17     image: postgres:15
18     container_name: app_postgres
19     environment:
20       POSTGRES_USER: usuario
21       POSTGRES_PASSWORD: clave123
22       POSTGRES_DB: basededatos
23     ports:
24       - "5432:5432"
25     volumes:
26       - pgdata:/var/lib/postgresql/data
27
28 volumes:
29   pgdata:
  
```

Probar ejecución de los servicios:
 docker-compose up --build



```

✓ Container app_postgres Created 0.7s
✓ Container app_python Created 0.2s
Attaching to app_postgres, app_python
app_postgres | The files belonging to this database system will be owned by user "postgres".
app_postgres | This user must also own the server process.
app_postgres |
app_postgres | The database cluster will be initialized with locale "en_US.utf8".
app_postgres | The default database encoding has accordingly been set to "UTF8".
app_postgres | The default text search configuration will be set to "english".
app_postgres |
app_postgres | Data page checksums are disabled.
app_postgres |
app_postgres | fixing permissions on existing directory /var/lib/postgresql/data ... ok
app_postgres | creating subdirectories ... ok
app_postgres | selecting dynamic shared memory implementation ... posix
app_postgres | selecting default max_connections ... 100
app_postgres | selecting default shared_buffers ... 128MB
app_python | Hola mundo desde Docker
app_postgres | selecting default time zone ... Etc/UTC
app_postgres | creating configuration files ... ok
app_python exited with code 0
app_postgres | running bootstrap script ... ok
app_postgres | performing post-bootstrap initialization ... ok
app_postgres | initdb: warning: enabling "trust" authentication for local connections
app_postgres | initdb: hint: You can change this by editing pg_hba.conf or using the option -A, or --auth-local and --auth-host, the next time
app_postgres | you run initdb.
app_postgres | syncing data to disk ... ok
app_postgres |
app_postgres | Success. You can now start the database server using:
app_postgres |
app_postgres | pg_ctl -D /var/lib/postgresql/data -l logfile start
  
```

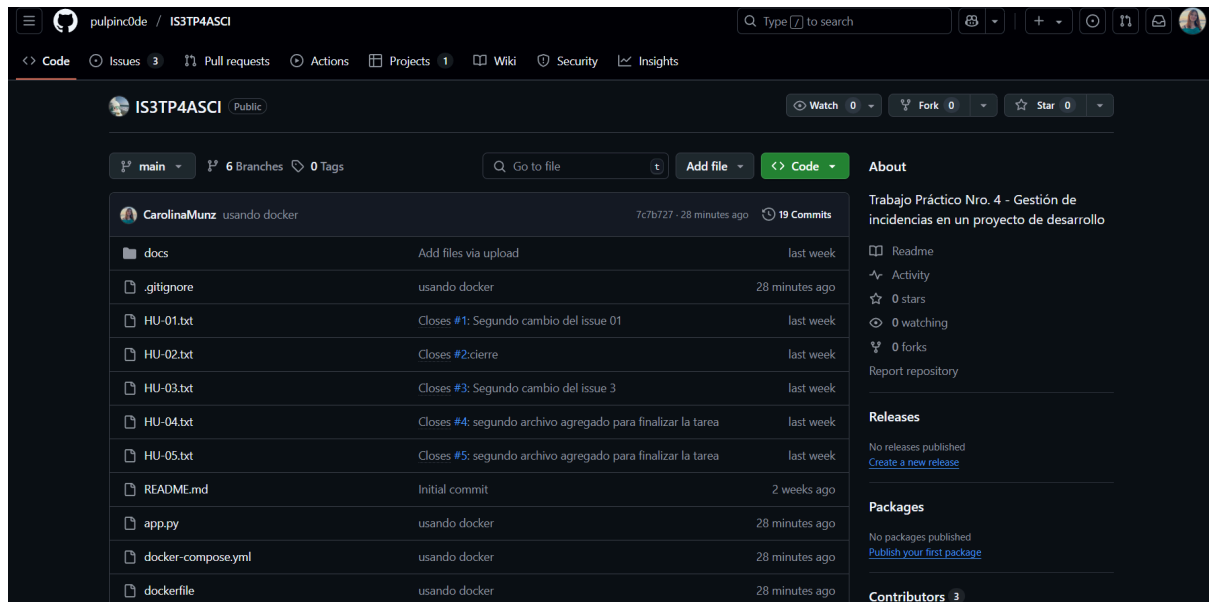
Ingeniería de Software III - Lic. en Sistemas de Información
Actualidad Informática - Analista en Sistemas de Computación
FCEQyN - UNAM

```

app_postgres | waiting for server to start....2025-06-18 02:47:16.281 UTC [49] LOG:  starting PostgreSQL 15.13 (Debian 15.13-1.pgdg120+1) on x
86_64-pc-linux-gnu, compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
app_postgres | 2025-06-18 02:47:16.285 UTC [49] LOG:  listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
app_postgres | 2025-06-18 02:47:16.296 UTC [52] LOG:  database system was shut down at 2025-06-18 02:47:15 UTC
app_postgres | 2025-06-18 02:47:16.308 UTC [49] LOG:  database system is ready to accept connections
app_postgres | done
app_postgres | server started
app_postgres | CREATE DATABASE
app_postgres |
app_postgres |
app_postgres | /usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/*
app_postgres |
app_postgres | 2025-06-18 02:47:16.641 UTC [49] LOG:  received fast shutdown request
app_postgres | waiting for server to shut down....2025-06-18 02:47:16.644 UTC [49] LOG:  aborting any active transactions
app_postgres | 2025-06-18 02:47:16.649 UTC [49] LOG:  background worker "logical replication launcher" (PID 55) exited with exit code 1
app_postgres | 2025-06-18 02:47:16.651 UTC [50] LOG:  shutting down
app_postgres | 2025-06-18 02:47:16.654 UTC [50] LOG:  checkpoint starting: shutdown immediate
app_postgres | 2025-06-18 02:47:16.794 UTC [50] LOG:  checkpoint complete: wrote 918 buffers (5.6%); 0 WAL file(s) added, 0 removed, 0 recycle
d; write=0.068 s, sync=0.066 s, total=0.144 s; sync files=301, longest=0.003 s, average=0.001 s; distance=4222 kB, estimate=4222 kB
app_postgres | 2025-06-18 02:47:16.820 UTC [49] LOG:  database system is shut down
app_postgres | done
app_postgres | server stopped
app_postgres |
app_postgres | PostgreSQL init process complete; ready for start up.
app_postgres |
app_postgres | 2025-06-18 02:47:16.911 UTC [1] LOG:  starting PostgreSQL 15.13 (Debian 15.13-1.pgdg120+1) on x86_64-pc-linux-gnu, compiled by
gcc (Debian 12.2.0-14) 12.2.0, 64-bit
app_postgres | 2025-06-18 02:47:16.913 UTC [1] LOG:  listening on IPv4 address "0.0.0.0", port 5432
app_postgres | 2025-06-18 02:47:16.913 UTC [1] LOG:  listening on IPv6 address ":::", port 5432
app_postgres | 2025-06-18 02:47:16.918 UTC [1] LOG:  listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
app_postgres | 2025-06-18 02:47:16.929 UTC [65] LOG:  database system was shut down at 2025-06-18 02:47:16 UTC
app_postgres | 2025-06-18 02:47:16.943 UTC [1] LOG:  database system is ready to accept connections
app_postgres | 2025-06-18 02:52:16.978 UTC [63] LOG:  checkpoint starting: time
app_postgres | 2025-06-18 02:52:21.133 UTC [63] LOG:  checkpoint complete: wrote 44 buffers (0.3%); 0 WAL file(s) added, 0 removed, 0 recycled

```

7. Documentar el proceso realizado con capturas de pantalla.

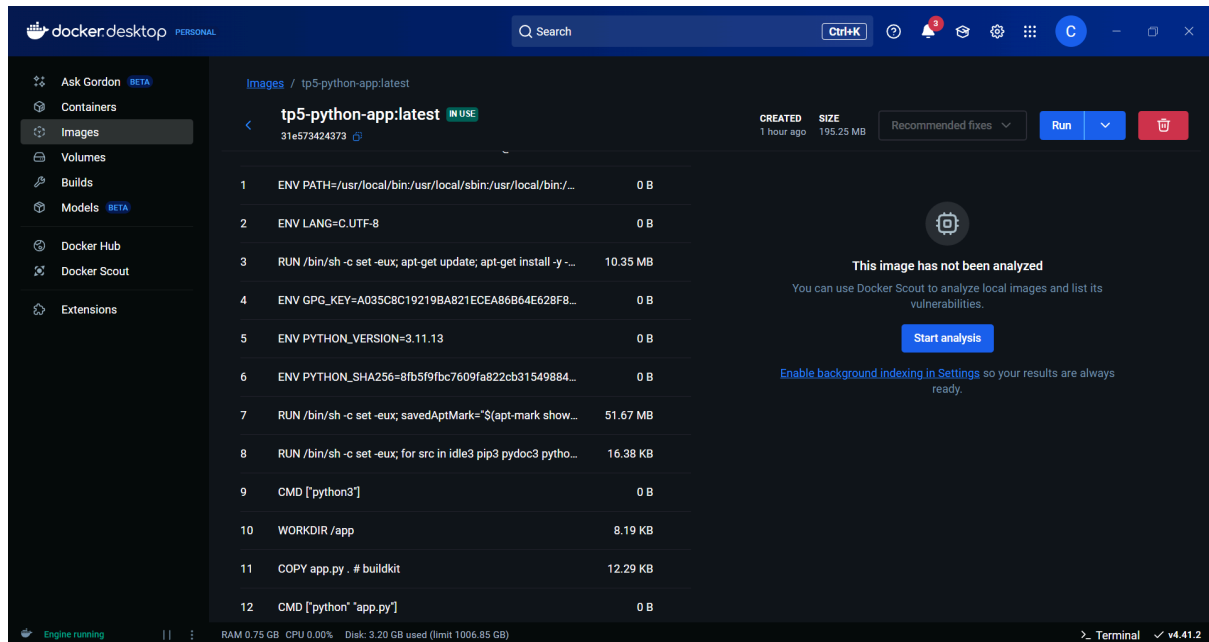


The screenshot displays a GitHub repository named **IS3TP4ASCI** (Public). The repository has 6 branches and 0 tags. The file list shows the following files and their commit history:

- docs**: Add files via upload (last week)
- .gitignore**: usando docker (28 minutes ago)
- HU-01.txt**: Closes #1: Segundo cambio del issue 01 (last week)
- HU-02.txt**: Closes #2: cierre (last week)
- HU-03.txt**: Closes #3: Segundo cambio del issue 3 (last week)
- HU-04.txt**: Closes #4: segundo archivo agregado para finalizar la tarea (last week)
- HU-05.txt**: Closes #5: segundo archivo agregado para finalizar la tarea (last week)
- README.md**: Initial commit (2 weeks ago)
- app.py**: usando docker (28 minutes ago)
- docker-compose.yml**: usando docker (28 minutes ago)
- dockerfile**: usando docker (28 minutes ago)

The right sidebar provides additional information about the repository, including the repository name, a description, and links to the README, Activity, Stars, and Forks.

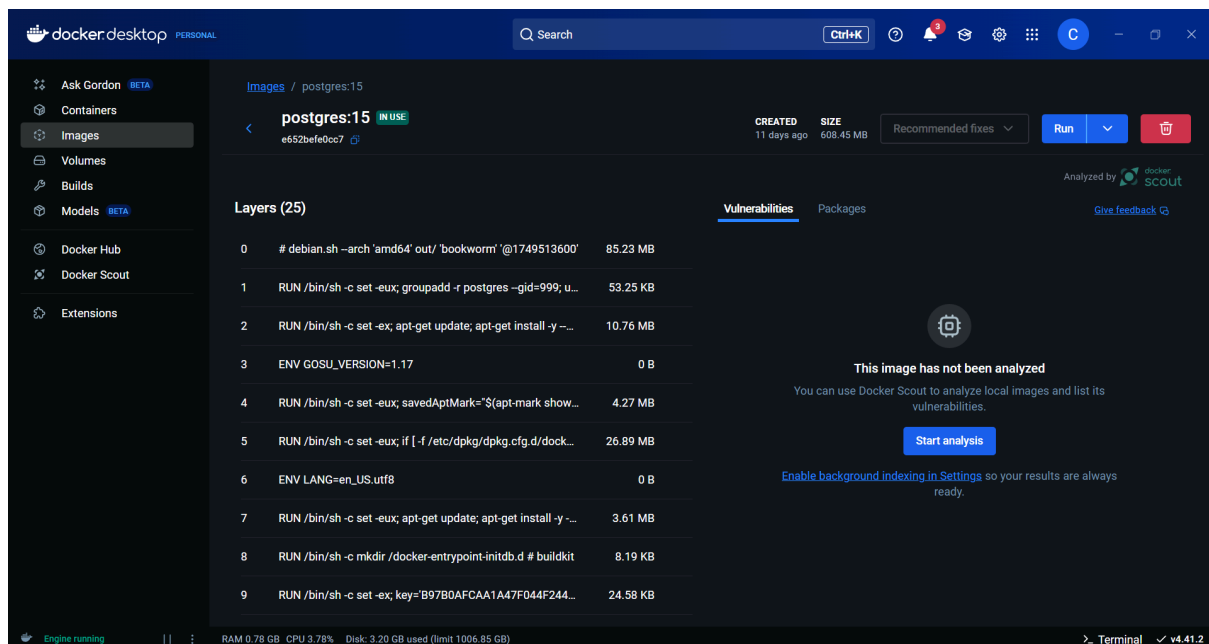
Ingeniería de Software III - Lic. en Sistemas de Información
Actualidad Informática - Analista en Sistemas de Computación
FCEQyN - UNaM



The screenshot shows the Docker Desktop interface for the 'tp5-python-app:latest' image. The image is 195.25 MB and was created 1 hour ago. It is currently 'IN USE'. The image layers are listed as follows:

Layer	Command	Size
1	ENV PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin/...	0 B
2	ENV LANG=C.UTF-8	0 B
3	RUN /bin/sh -c set -eux; apt-get update; apt-get install -y ...	10.35 MB
4	ENV GPG_KEY=A035C8C19219BA821ECEA86B64E628F8...	0 B
5	ENV PYTHON_VERSION=3.11.13	0 B
6	ENV PYTHON_SHA256=8fb5f9fbc7609fa822cb31549884...	0 B
7	RUN /bin/sh -c set -eux; savedAptMark="\$(apt-mark show...	51.67 MB
8	RUN /bin/sh -c set -eux; for src in idle3 pip3 pydoc3 pytho...	16.38 KB
9	CMD ["python3"]	0 B
10	WORKDIR /app	8.19 KB
11	COPY app.py . # buildkit	12.29 KB
12	CMD ["python" "app.py"]	0 B

The right panel indicates that the image has not been analyzed by Docker Scout. A 'Start analysis' button is available. A message suggests enabling background indexing in settings for better results.



The screenshot shows the Docker Desktop interface for the 'postgres:15' image. The image is 608.45 MB and was created 11 days ago. It is currently 'IN USE'. The image layers are listed as follows:

Layer	Command	Size
0	# debian.sh --arch 'amd64' out/ 'bookworm' '@1749513600'	85.23 MB
1	RUN /bin/sh -c set -eux; groupadd -r postgres --gid=999; u...	53.25 KB
2	RUN /bin/sh -c set -ex; apt-get update; apt-get install -y ...	10.76 MB
3	ENV GOSU_VERSION=1.17	0 B
4	RUN /bin/sh -c set -eux; savedAptMark="\$(apt-mark show...	4.27 MB
5	RUN /bin/sh -c set -eux; if [-f /etc/dpkg/dpkg.cfg.d/dock...	26.89 MB
6	ENV LANG=en_US.utf8	0 B
7	RUN /bin/sh -c set -eux; apt-get update; apt-get install -y ...	3.61 MB
8	RUN /bin/sh -c mkdir /docker-entrypoint-initdb.d # buildkit	8.19 KB
9	RUN /bin/sh -c set -ex; key='B97B0AFCAA1A47F044F244...	24.58 KB

The right panel indicates that the image has not been analyzed by Docker Scout. A 'Start analysis' button is available. A message suggests enabling background indexing in settings for better results.

Repositorio trabajado:

[pulpinc0de/IS3TP4ASCI](https://github.com/pulpinc0de/IS3TP4ASCI): Trabajo Práctico Nro. 4 - Gestión de incidencias en un proyecto de desarrollo