

#### **Grupo No.4:**

Elda Magally Calderon Motta	16003182
Ligia Deyanira Aguilar Estrada	19004789
Victor Rene Pérez Mayen	19005180

### **PRODUCT DEVELOPMENT**

#### **PROYECTO FINAL: COVID-19**

#### **1. Resumen**

El presente proyecto consiste en poder visualizar la cantidad de casos confirmados, casos fallecidos y casos recuperados del virus COVID-19. Para ello se utilizó un dashboard de Shiny Apps creado en un contenedor de Docker, este dashboard visualiza la data almacenada en una Base de Datos de MySQL ubicada también en un contenedor de Docker; dicha base de datos recibe una ingesta de datos por medio de Airflow, ubicado en un contenedor de Docker, el cual sirve como un ETL para la base de datos. Se obtuvo como resultado una visualización a nivel global de los casos de COVID-19, el cual puede recibir información de manera dinámica gracias a los DAGS de Airflow.

#### **2. Introducción**

Docker es una aplicación que simplifica el proceso de gestionar los procesos de aplicaciones en contenedores. Los contenedores le permiten ejecutar sus aplicaciones en procesos aislados de recursos. Se parecen a las máquinas virtuales, sin embargo, los contenedores son más portátiles, tienen más recursos y son más dependientes del sistema operativo host.

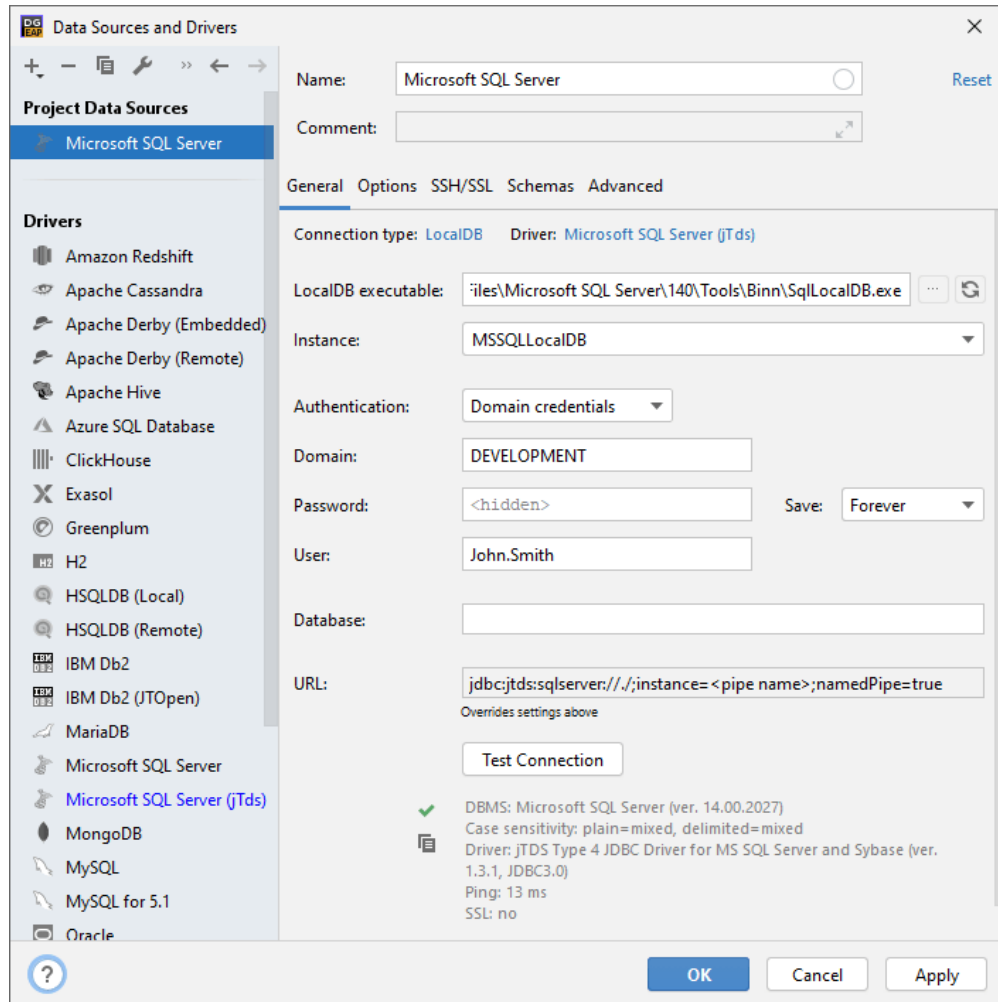
#### **3. Desarrollo**

##### **3.1. Instalación de MySQL en contenedores**

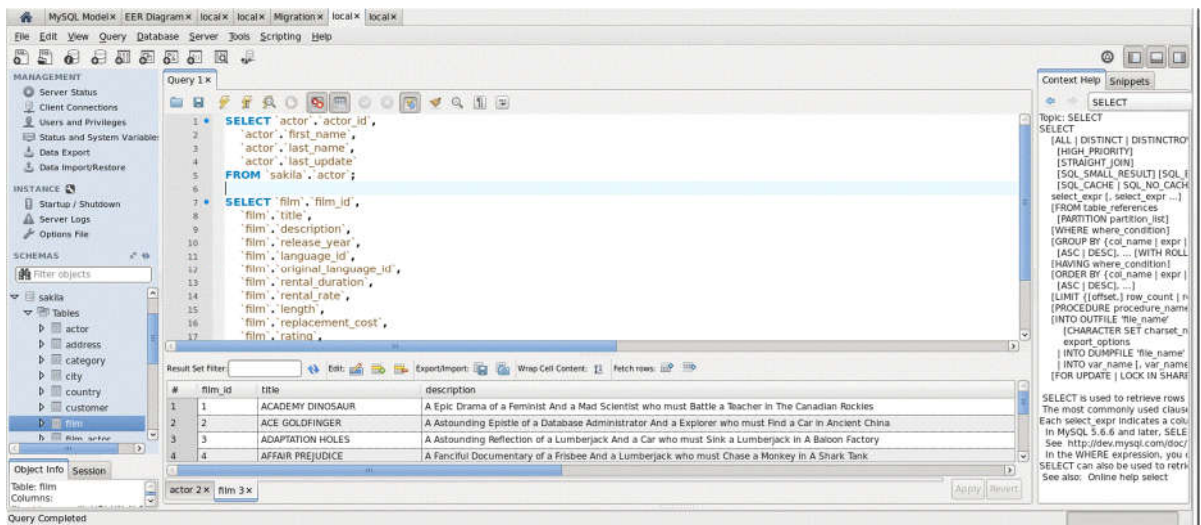
MySQL es un sistema de gestión de bases de datos de código abierto. Gestiona sus datos usando una base de datos relacional y SQL (Lenguaje de consulta estructurada).

### 3.2. Conexión a la base de datos

Para la conexión de la base de datos con los contenedores, se procedió a utilizar DataGrip. Se configuró la conexión para MySQL, con valores similares a los que se muestran a continuación:



Para poder acceder a la base de datos se utilizó MySQL Workbench utilizando los parámetros de conexión que se indicaron al momento de crear



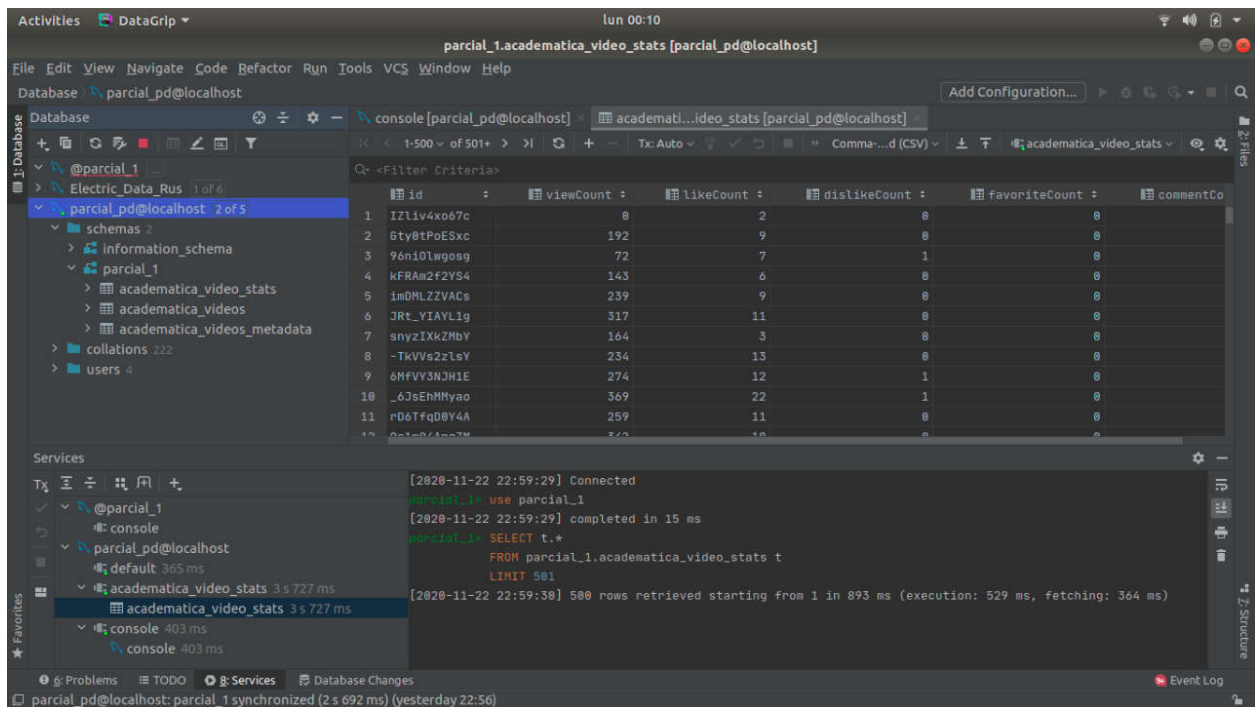
### 3.3. Ingesta de información para la base de datos

Por medio de los DAGS de Airflow, se realiza el proceso de ETL hacia la base de datos. Gracias a este proceso, la información de los casos confirmados, recuperados y fallecidos se cargan a la base de datos.

The screenshot shows the Airflow web interface. The 'DAGs' tab is selected. A search bar is at the top right. Below it, a table lists the DAGs. The table has columns: DAG, Schedule, Owner, Recent Tasks, Last Run, DAG Runs, and Links. Two DAGs are listed: 'predict\_model\_dag' and 'train\_model\_dag'. The 'predict\_model\_dag' is scheduled to run every 1 day at 00:00:00. The 'train\_model\_dag' has no schedule. Both DAGs are owned by 'airflow' and have 1 recent task each. The 'predict\_model\_dag' was last run on 2020-09-22 13:59 and has 1 DAG run. The 'train\_model\_dag' was last run on 2020-09-23 13:59 and has 1 DAG run. The interface shows 1 of 2 entries.

DAG	Schedule	Owner	Recent Tasks	Last Run	DAG Runs	Links
<a href="#">predict_model_dag</a>	1 day, 00:00:00	airflow	1	2020-09-22 13:59	1	<a href="#">View</a> <a href="#">Refresh</a> <a href="#">Delete</a> <a href="#">Clone</a> <a href="#">Copy</a> <a href="#">Share</a>
<a href="#">train_model_dag</a>	None	airflow	1	2020-09-23 13:59	1	<a href="#">View</a> <a href="#">Refresh</a> <a href="#">Delete</a> <a href="#">Clone</a> <a href="#">Copy</a> <a href="#">Share</a>

Showing 1 to 2 of 2 entries



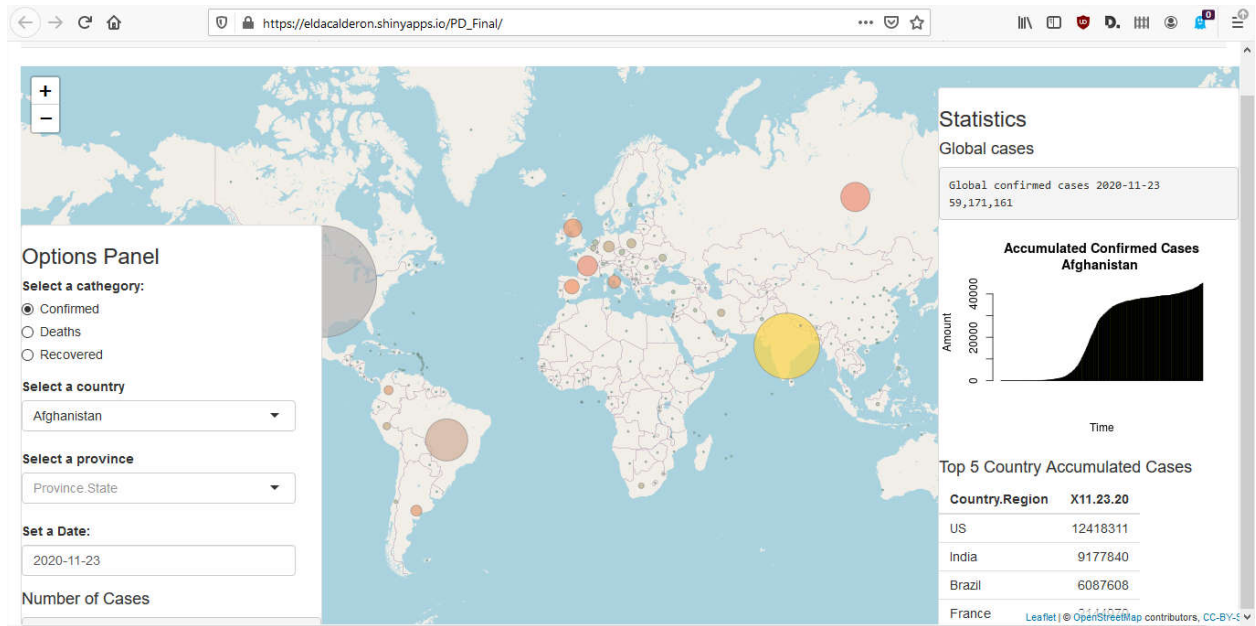
### 3.4. Configuración con R

En el apartado de R se creo un dashboard por medio de Shiny Apps, este dashboard visualiza la cantidad de casos mencionados anteriormente por País, región y por fecha. En el dashboard también se pueden visualizar la cantidad de casos por medio de una gráfica acumulativa, y los 5 países con más casos por fechas específicas.

Se comparte el link del dashboard en Shiny APP Server:

[https://eldacalderon.shinyapps.io/PD\\_Final/](https://eldacalderon.shinyapps.io/PD_Final/)

Y por último, se muestra como es la visualización del dashboard:



#### 4. Conclusiones

- Se instaló Docker, se trabajó con imágenes y contenedores haciendo push del material visto en el curso.
- Ahora se tiene una configuración de MySQL para base de datos.
- Se realizó una conexión de la base de datos con los contenedores de Docker usando DataGrip
- Se hizo la conexión de R server con la base de datos, ambos levantados en Docker container.
- Se demostró que Airflow es permite el proceso de ETL hacia la base de datos.

#### 5. Recomendaciones

- Se recomienda utilizar contenedores de Docker para base de datos MySQL o SQL Server, ya que son los que tienen mayor cantidad de documentación.
- Es recomendable corroborar la correcta conexión entre los dashboards con los contenedores, para que la data sea visualizada en tiempo real.
- Se debe tener especial cuidado en la configuración de los DAGS para que no tenga errores el proceso de ETL en los contenedores.

## 6. Justificación

El presente Proyecto se realizó con el fin de poder realizar una conexión entre los DAGS de Airflow con una base de datos con contenedores hechos en Docker, y que los datos se puedan visualizar de una manera “amigable” al usuario utilizando dashboards elaborados en shiny apps de R. El fin principal es que el usuario pueda visualizar de manera asequible por medio de una virtualización óptima gracias a Docker contains los casos recuperados, confirmados y fallecidos a causa del virus de COVID-19.