***AWS IoT TwinMaker :***

***Use case presentation :***

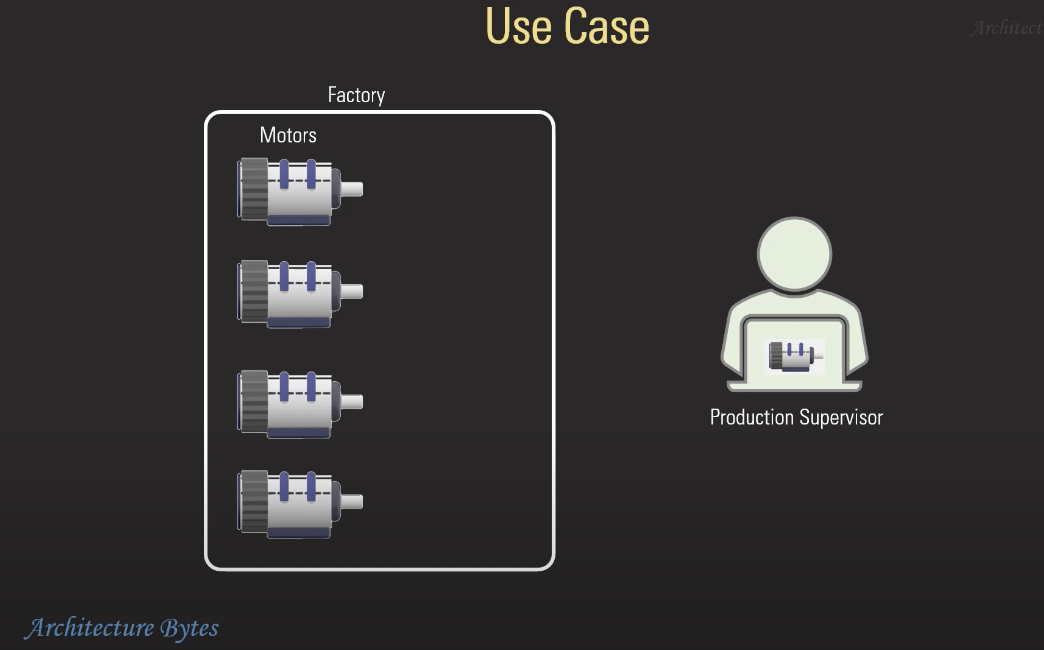
This is a tutorial related to AWS (Amazon Web Services) IoT TwinMaker.

we suggest a use case within a factory setting where there are multiple motors running on the production floor.

A production supervisor is looking to monitor these motors, likely for efficiency, maintenance, and operational purposes.

The scenario is focused on the supervisor's need to access a visual representation of the historical data of the motors, specifically tracking their speeds.

This use case exemplifies the benefits of IoT and digital twins in industrial environments, where data-driven insights lead to improved operational efficiency, reduced costs, and enhanced equipment lifespan.



***Architecture:***

The architecture is a streamlined process for collecting, processing, and visualizing data from an industrial motor within a factory.

Here's an analysis of the flow and components involved:

**Factory Level –**

**Motor**: It all starts with a motor in the factory.

The motor is likely equipped with sensors that collect various data points such as speed, temperature, vibration, etc.

**Data Collection to AWS Cloud –**

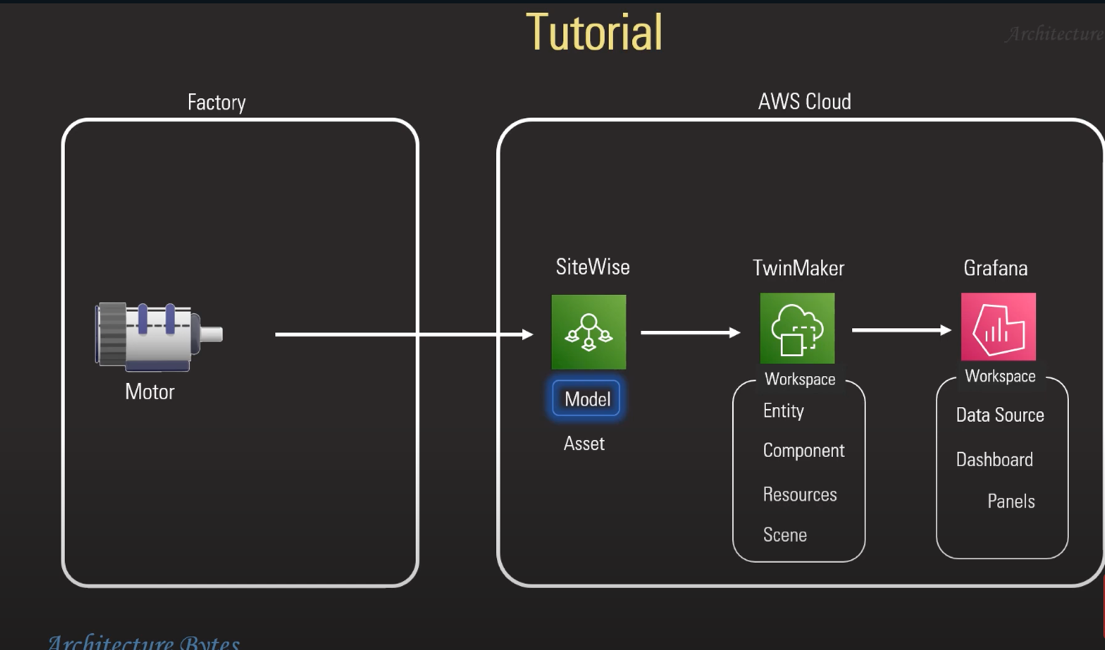
**SiteWise**: The data from the motor is then transmitted to AWS Cloud, where AWS IoT SiteWise is used as the first point of interaction. AWS IoT SiteWise is a managed service that collects, organizes, and analyzes data from industrial equipment at scale. It provides the capability to structure and label data coming from the factory floor, making it easier to identify and work with.

**Data Processing and Digital Twin Creation –**

**TwinMaker**: The processed data flows into AWS IoT TwinMaker, which is used to create a digital twin of the motor. A digital twin is a virtual representation of a physical object or system. This step involves enriching the raw data with contextual information to create a comprehensive virtual model of the motor that reflects its real-time status and historical behavior.

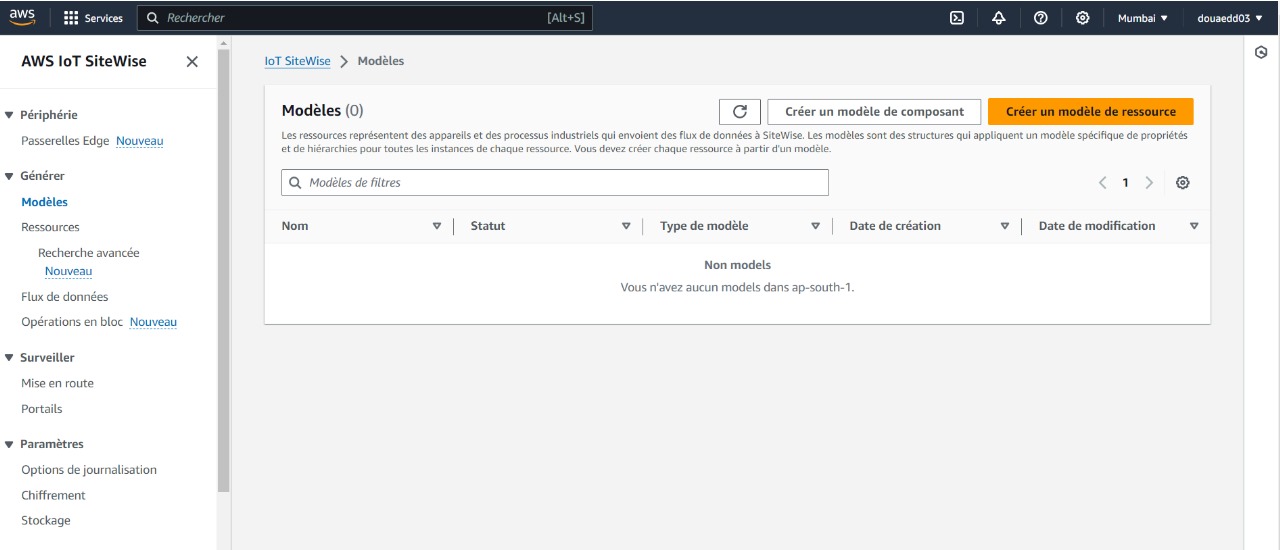
**Data Visualization –**

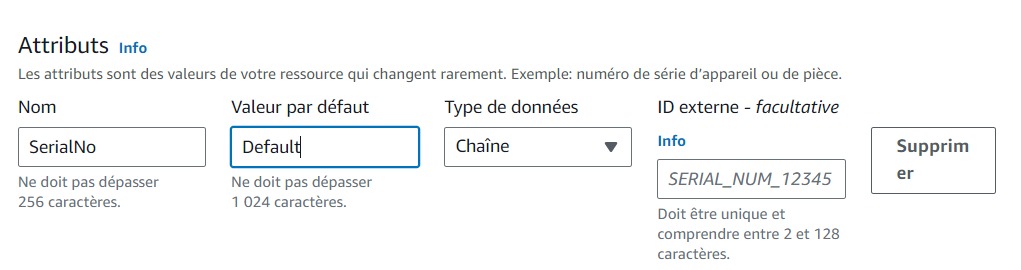
**Grafana**: Finally, the data is piped into Grafana, which is an open-source platform for monitoring and observability. Grafana takes the enriched data from TwinMaker and uses it to create visual dashboards. These dashboards are designed to make it easy for the production supervisor or other stakeholders to monitor the motor's performance, analyze historical data, and make informed decisions based on the insights provided.

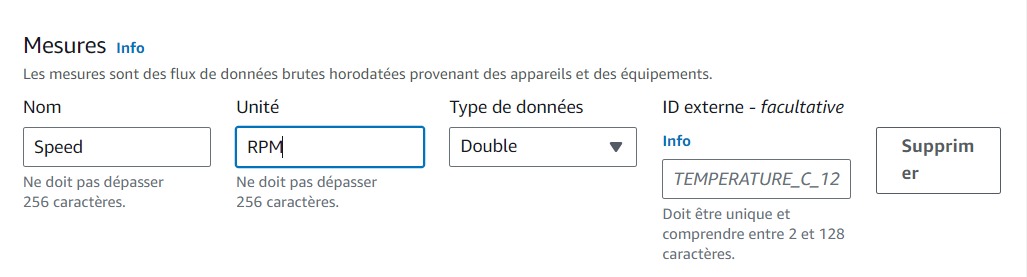


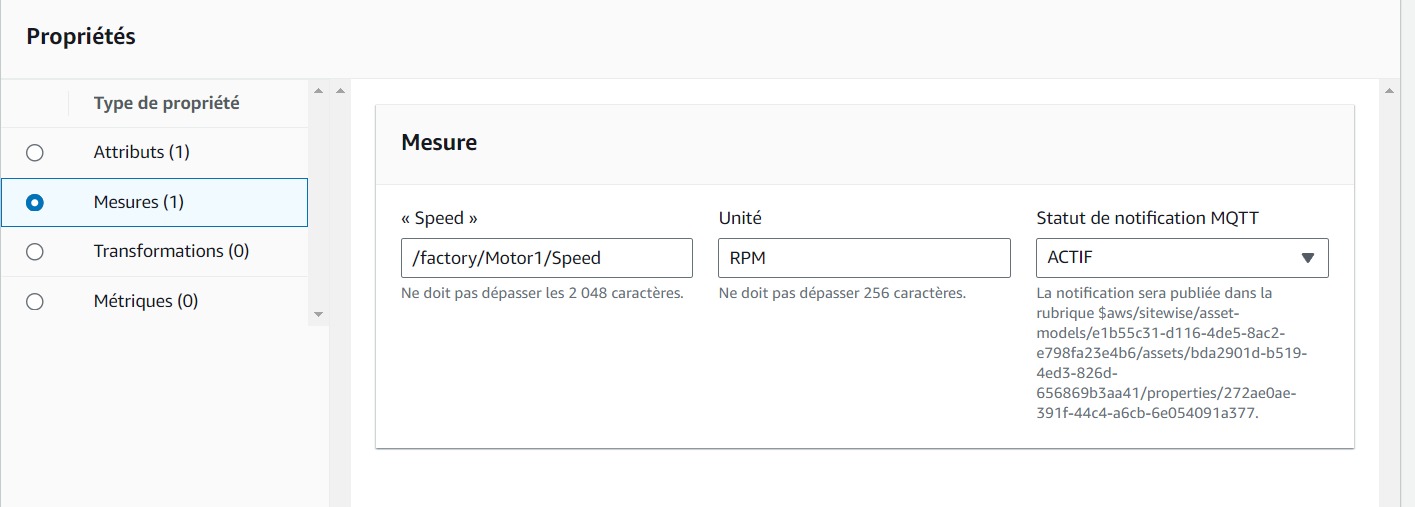
1. SiteWise :

* Create Model

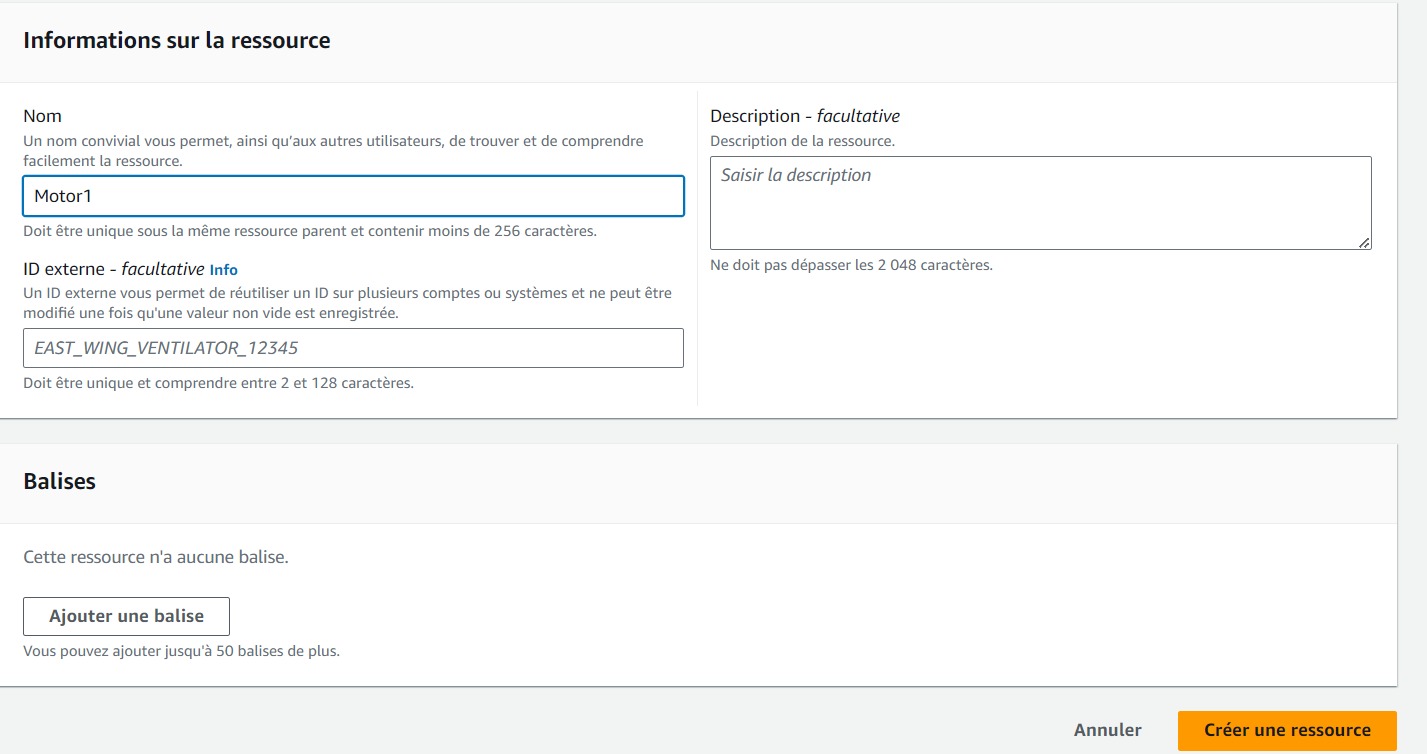


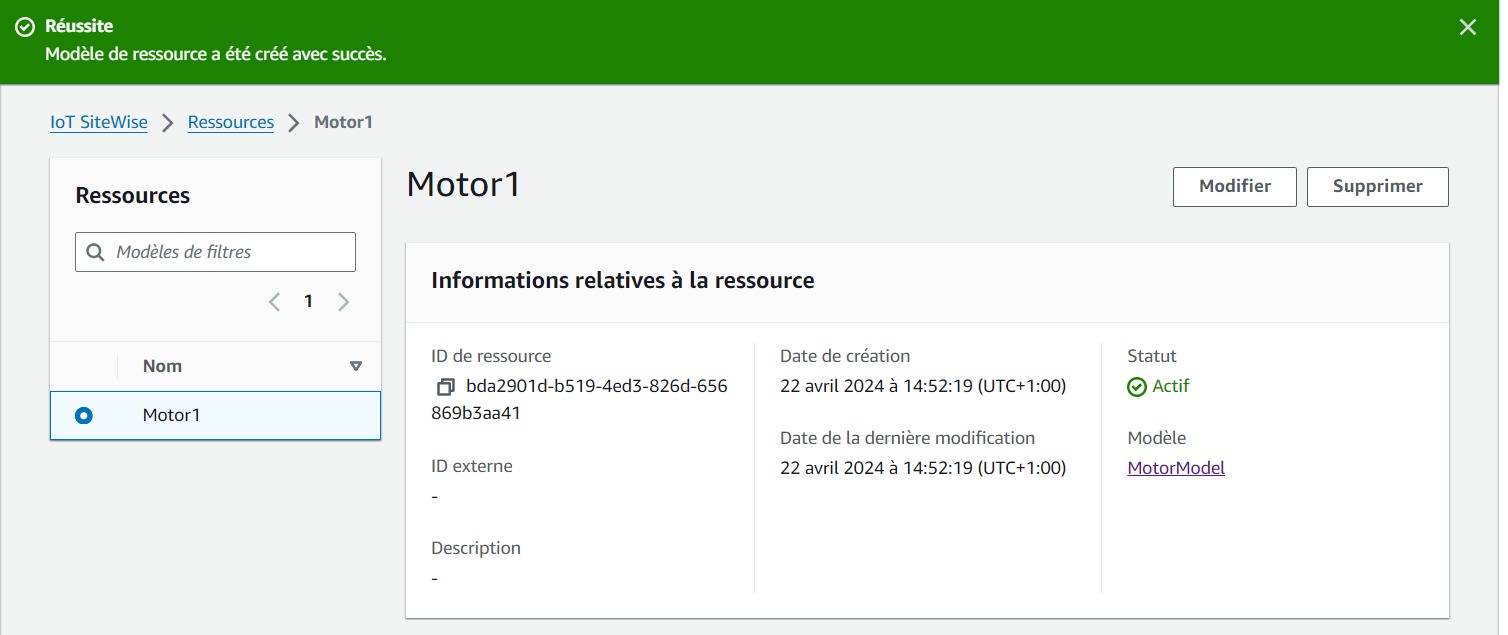






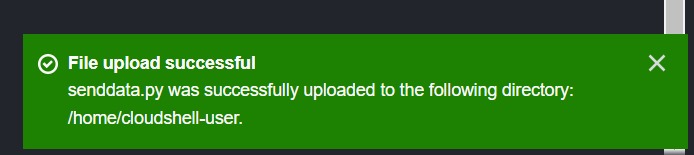
* Asset creation :

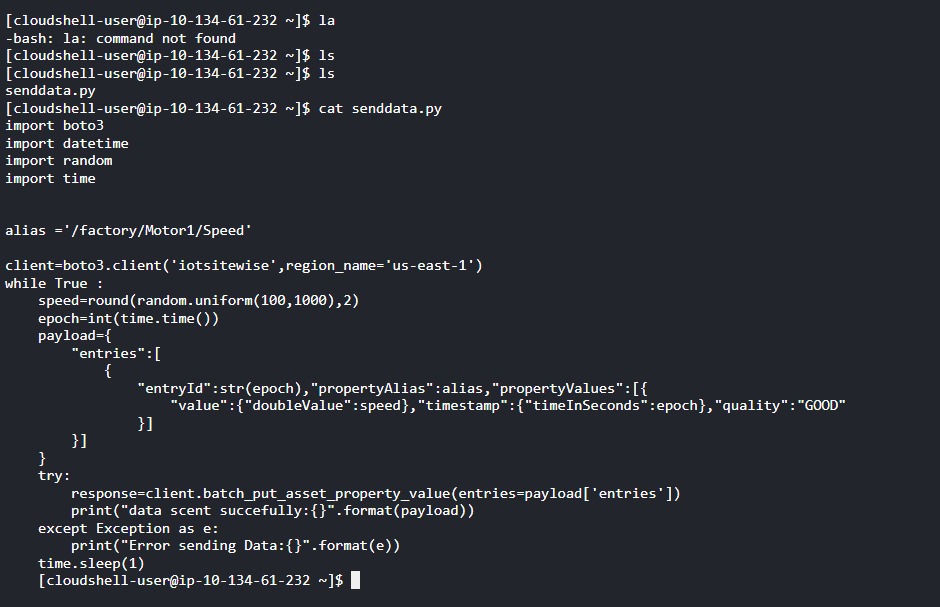


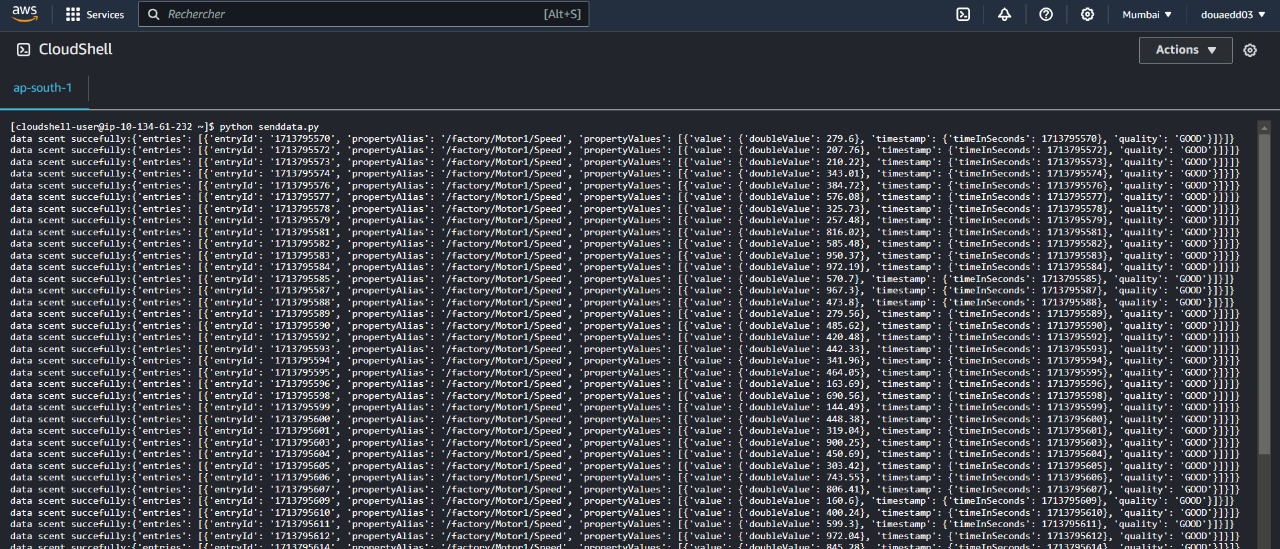


* Send Data:

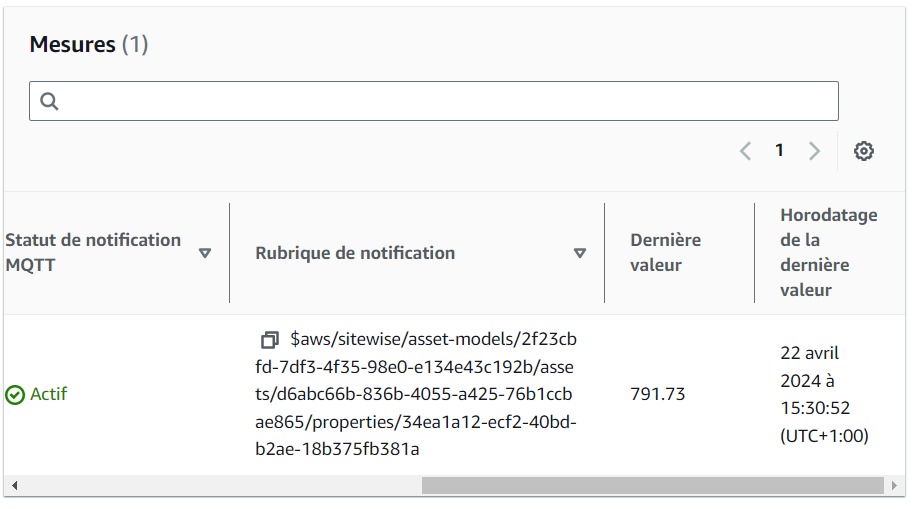


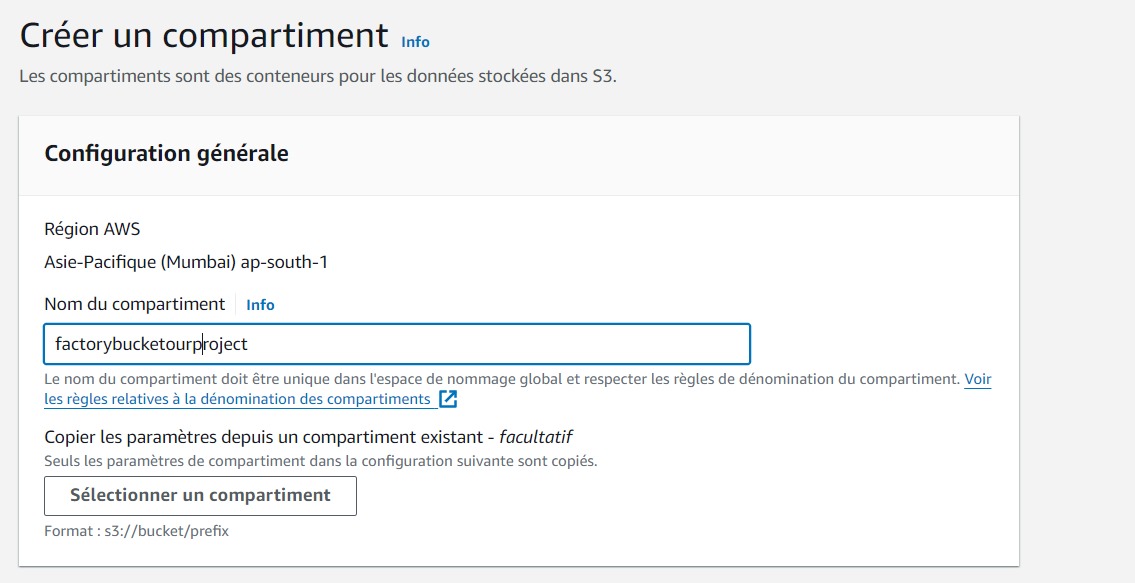


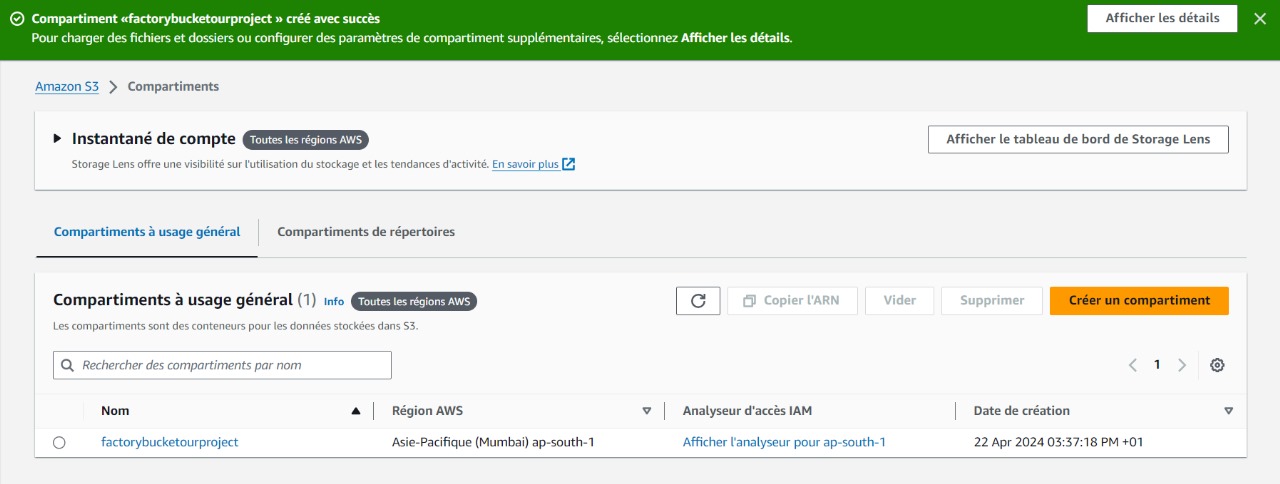




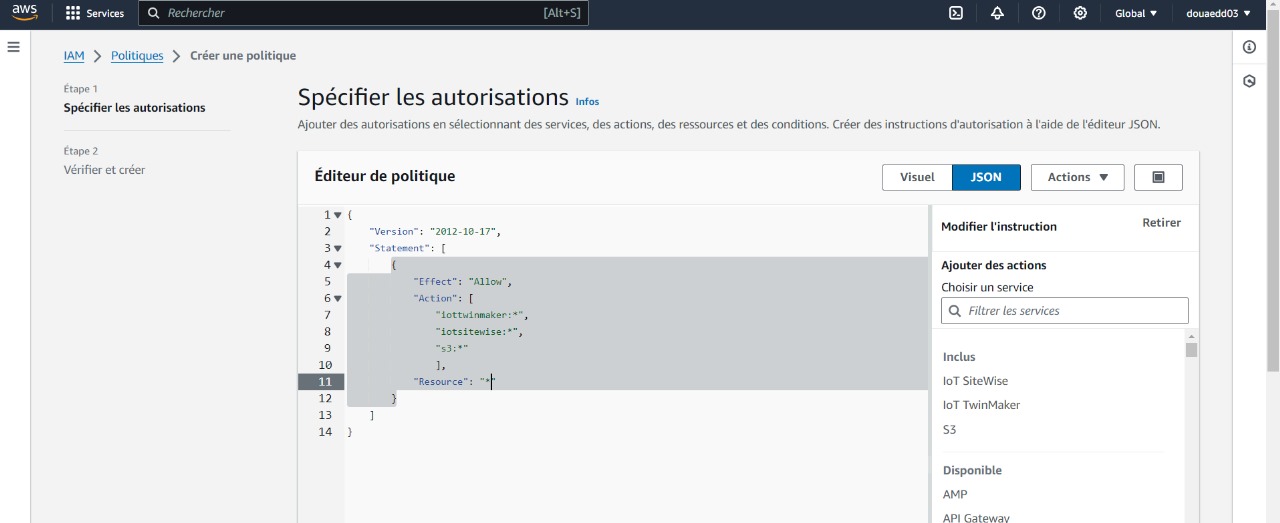
1. TWIN MAKER :
2. CREATE WORKSPACE:

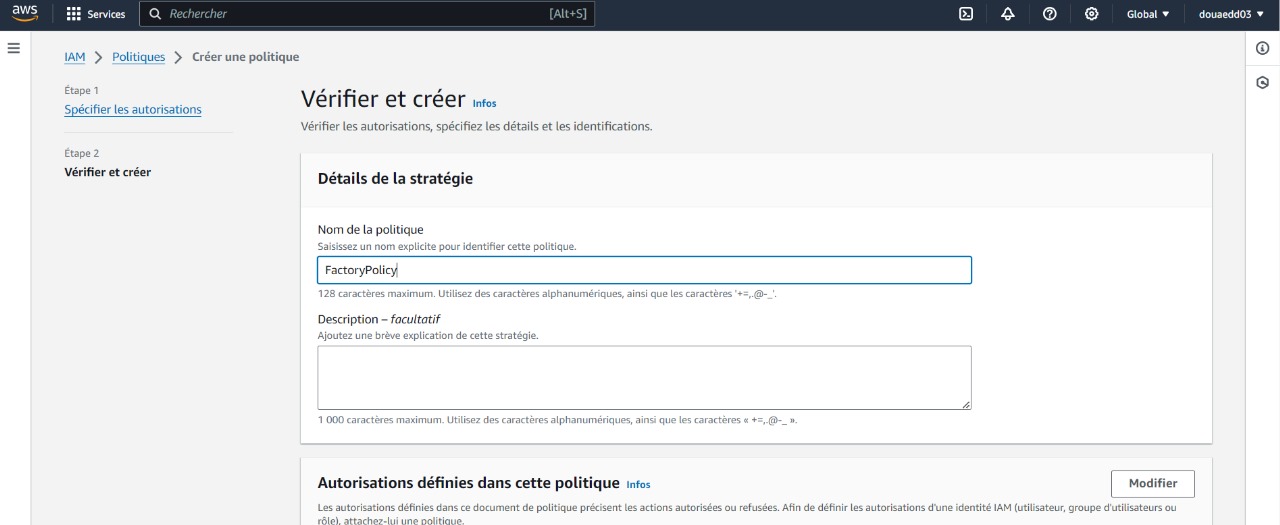


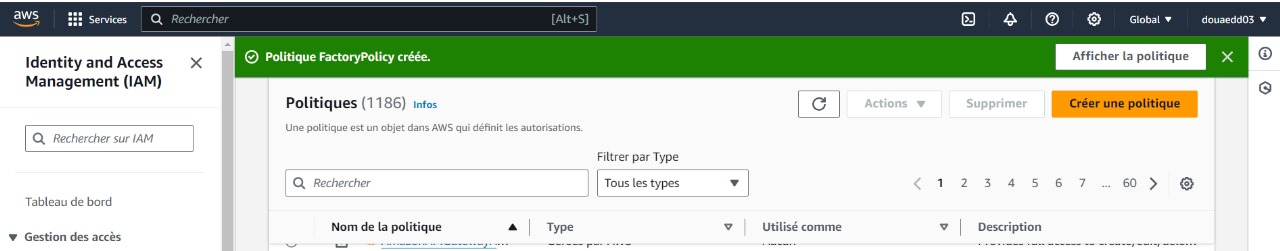




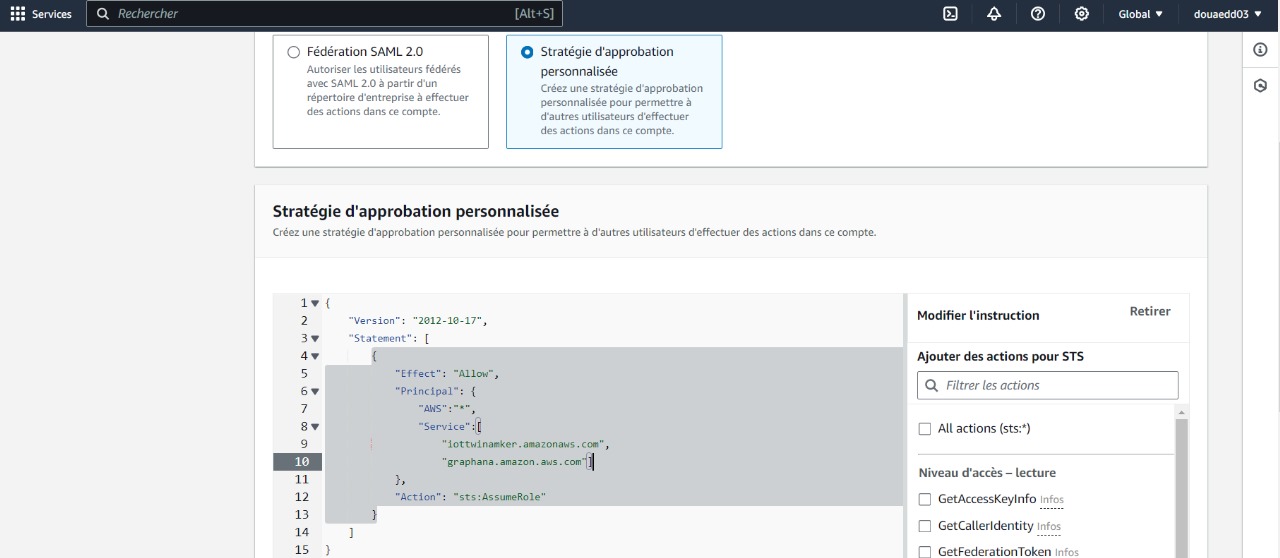
* CREATING POLICY :

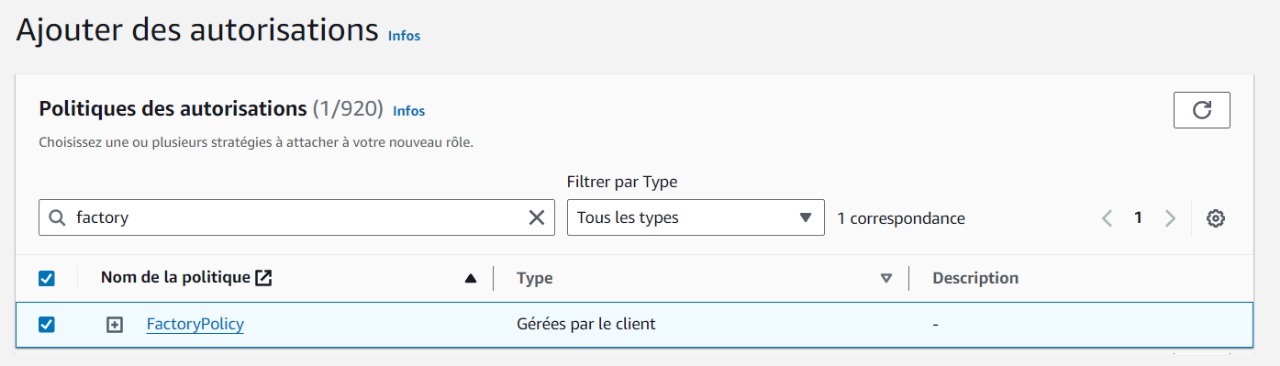




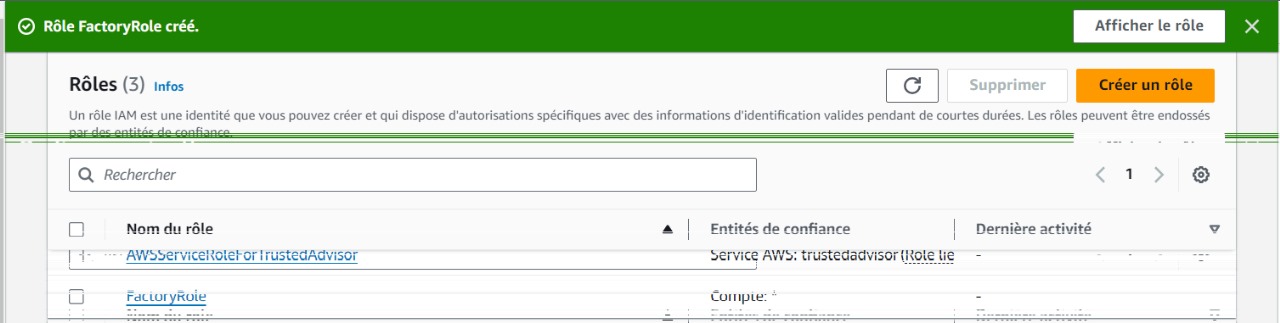


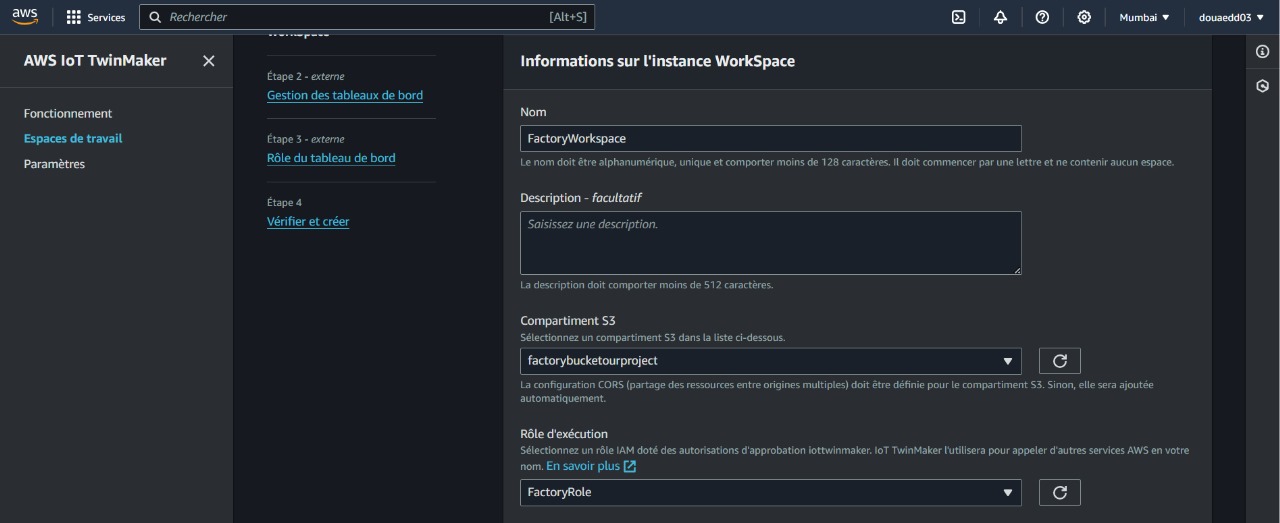
* CREATING ROLE :

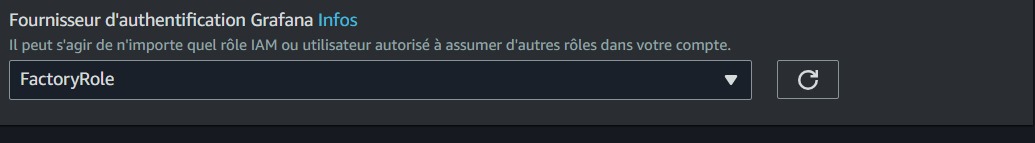


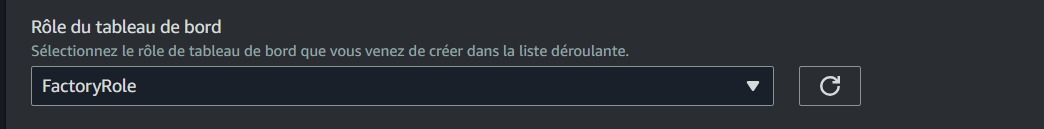


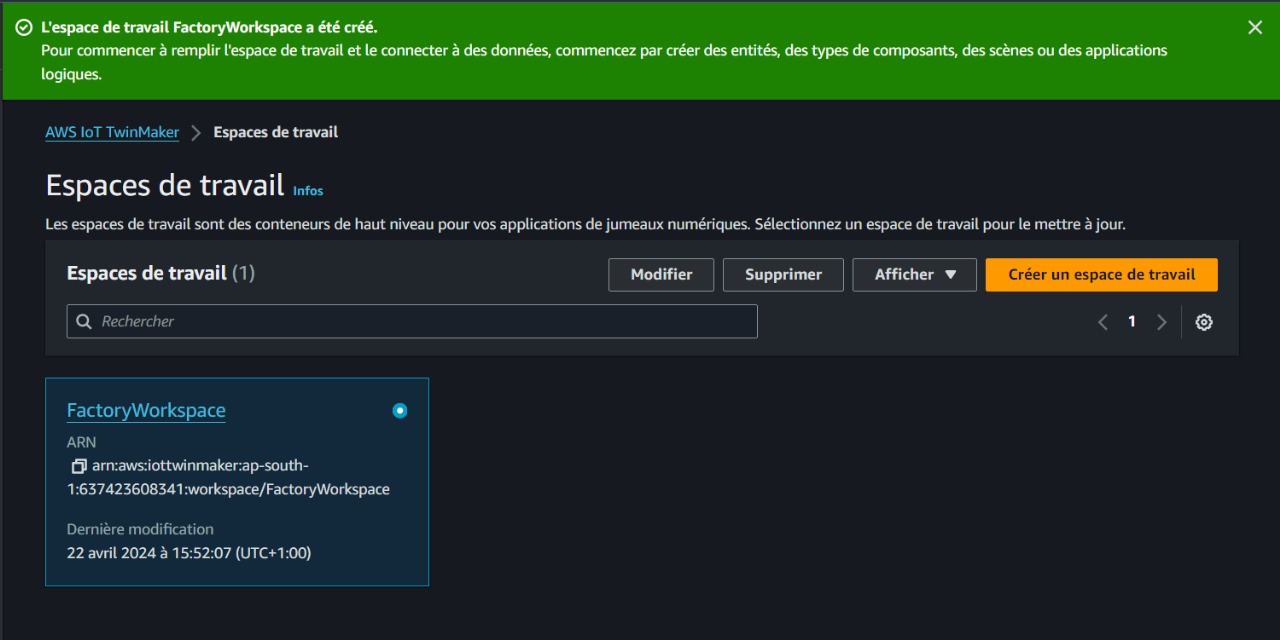










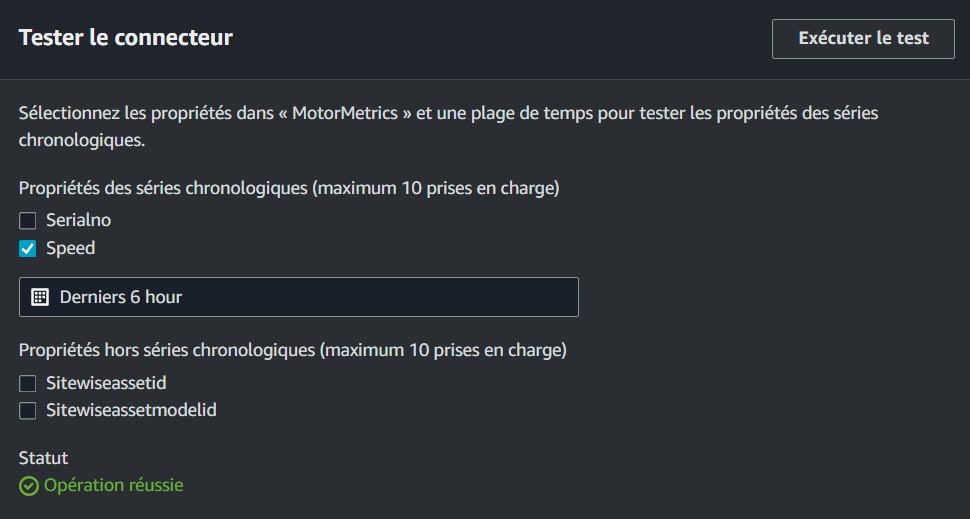


1. Create Entity:

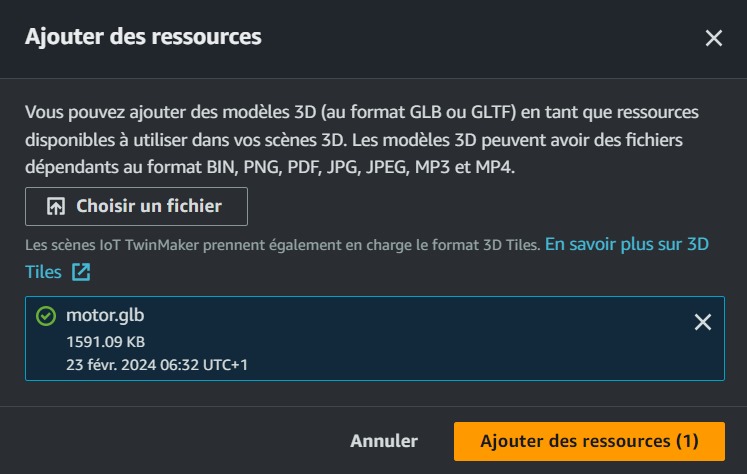


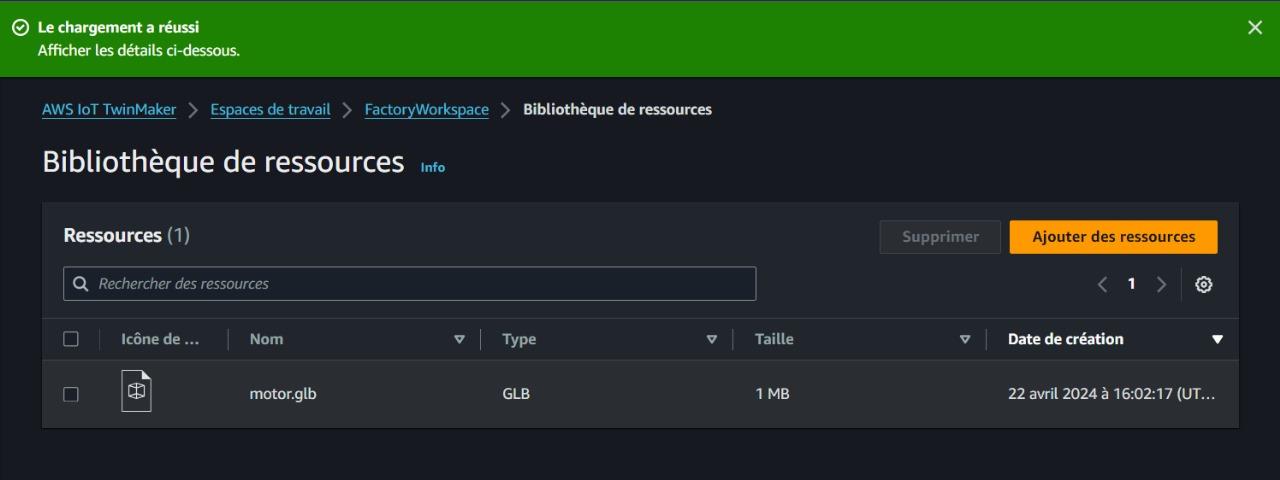
* Adding component:





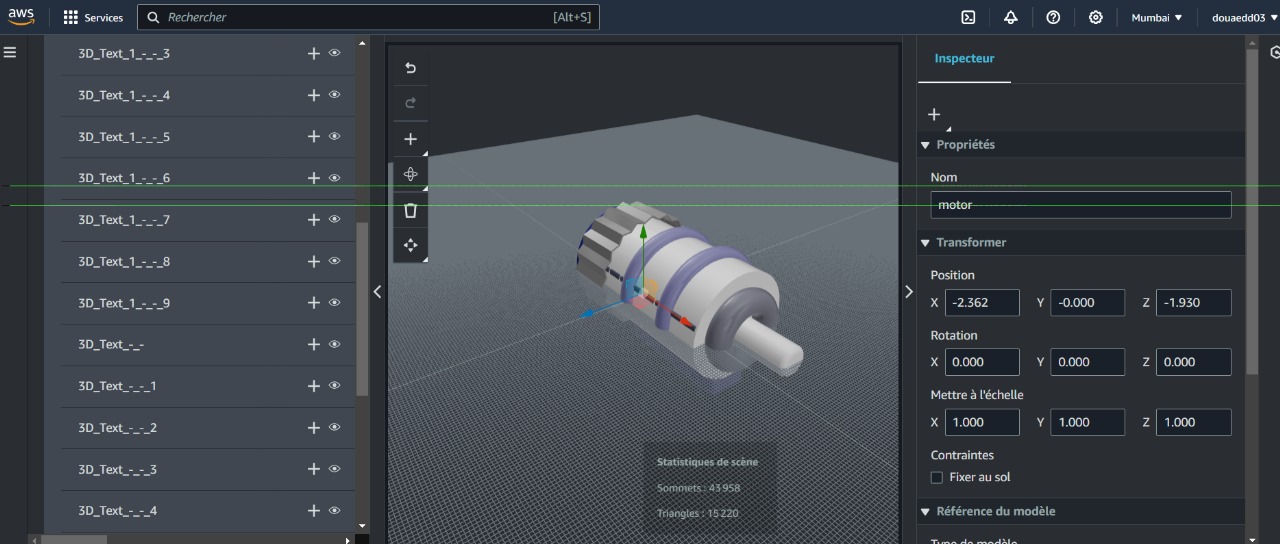
1. Add Resources :

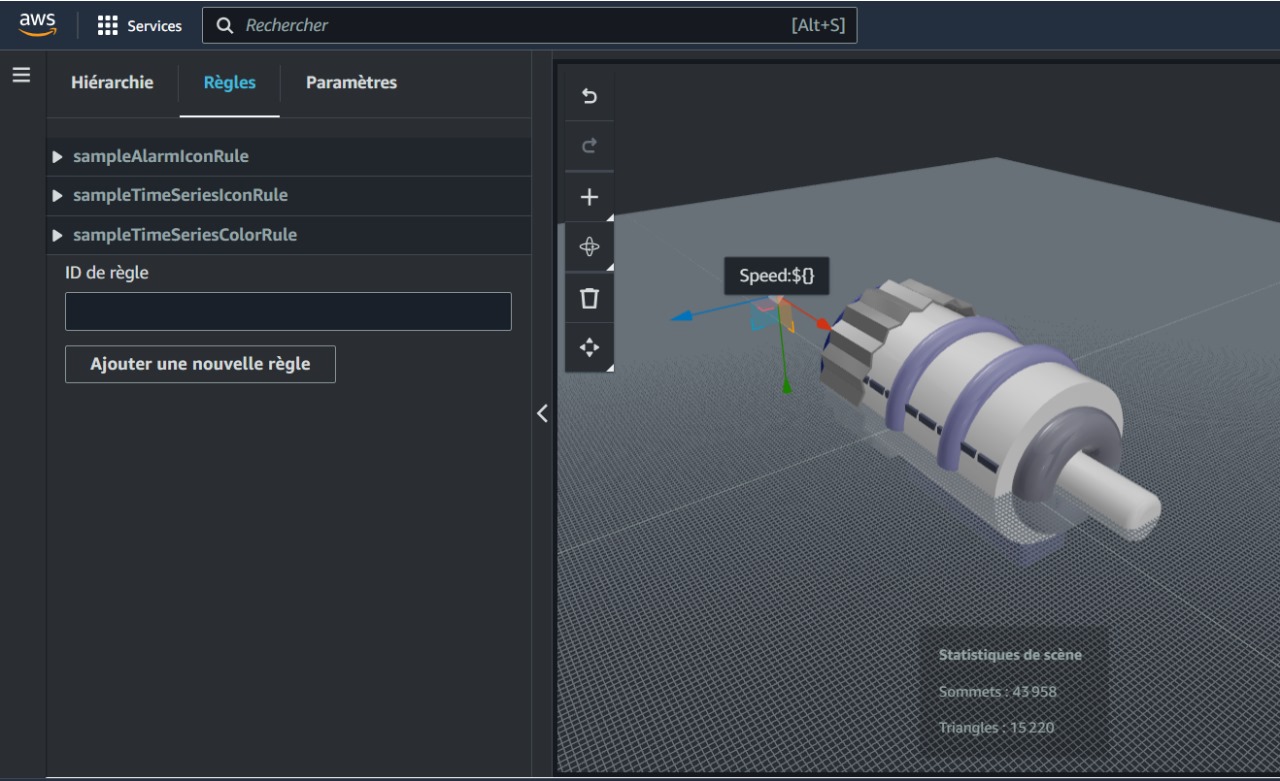




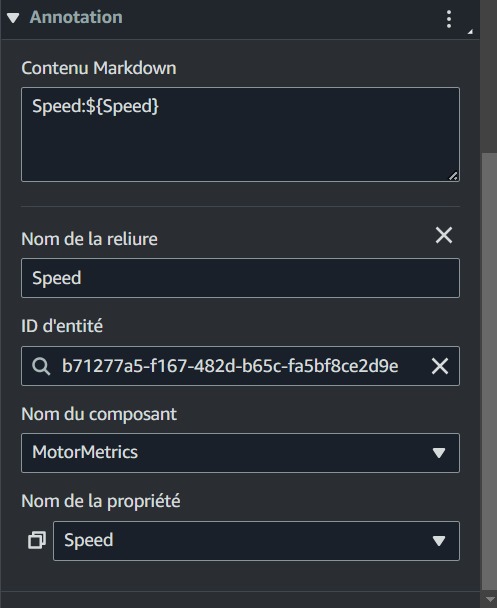
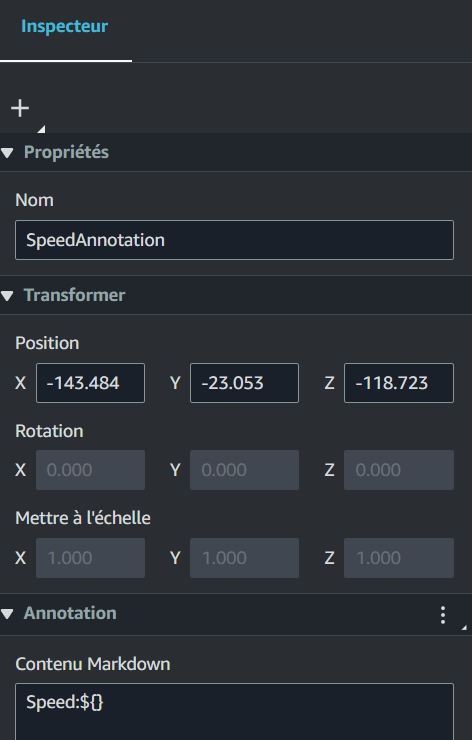
1. Creating Factory Scene:

* Add Motor Model:

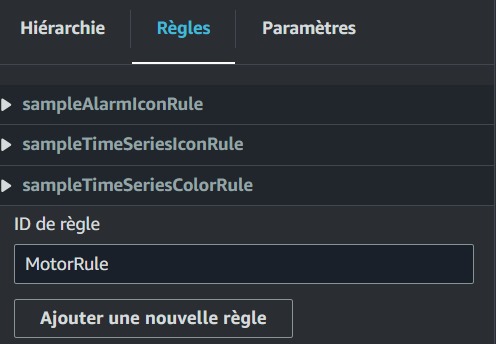


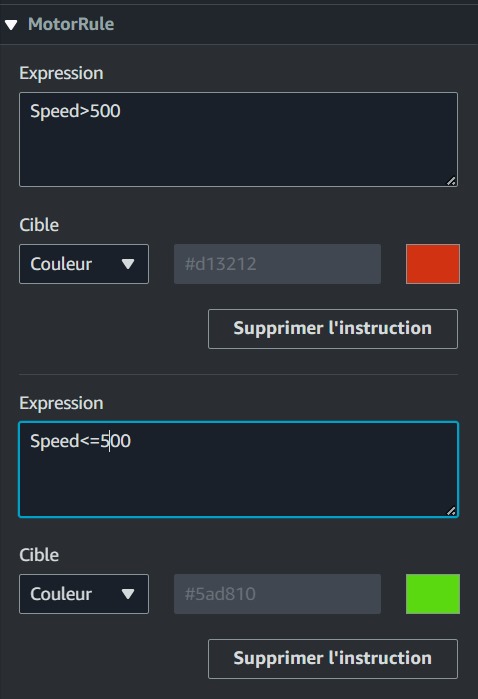


* Add Speed Annotation:

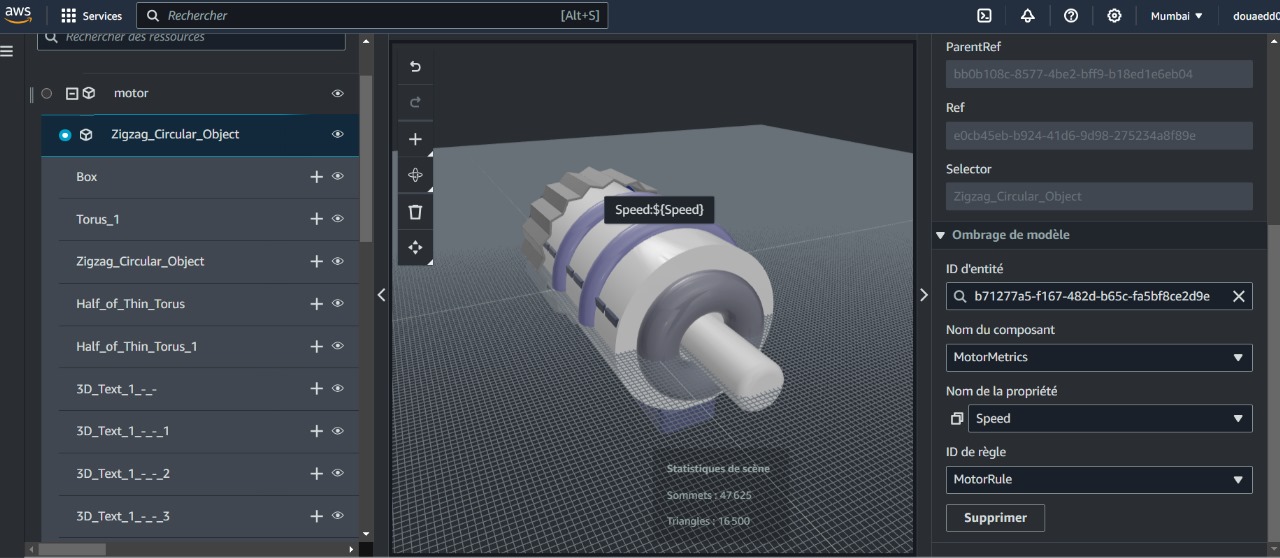


* Add Motor Rule:



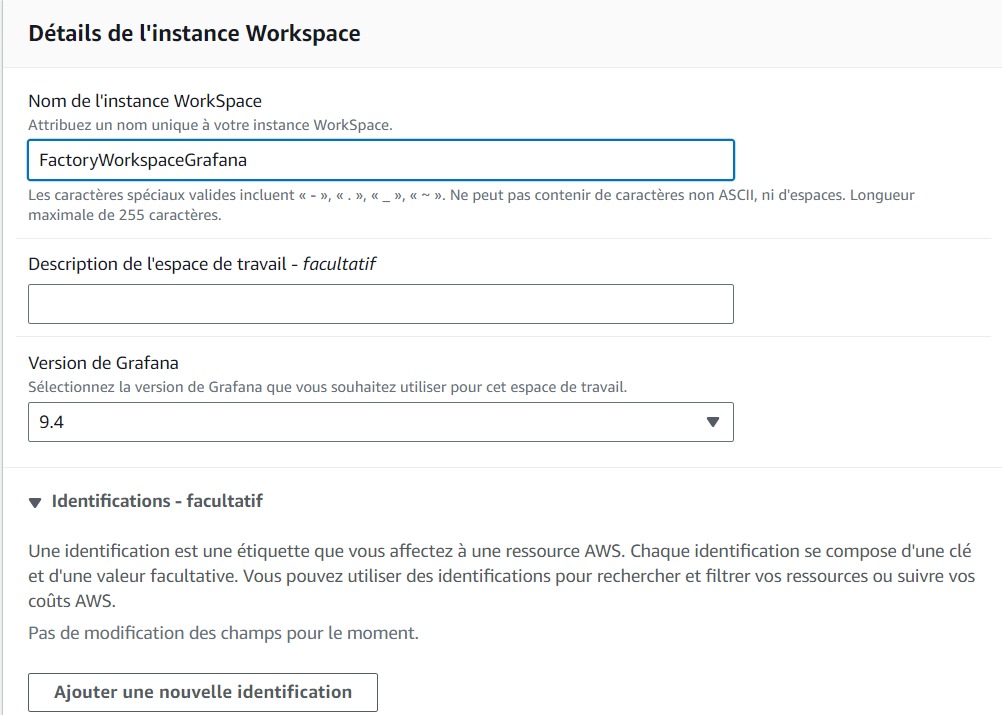


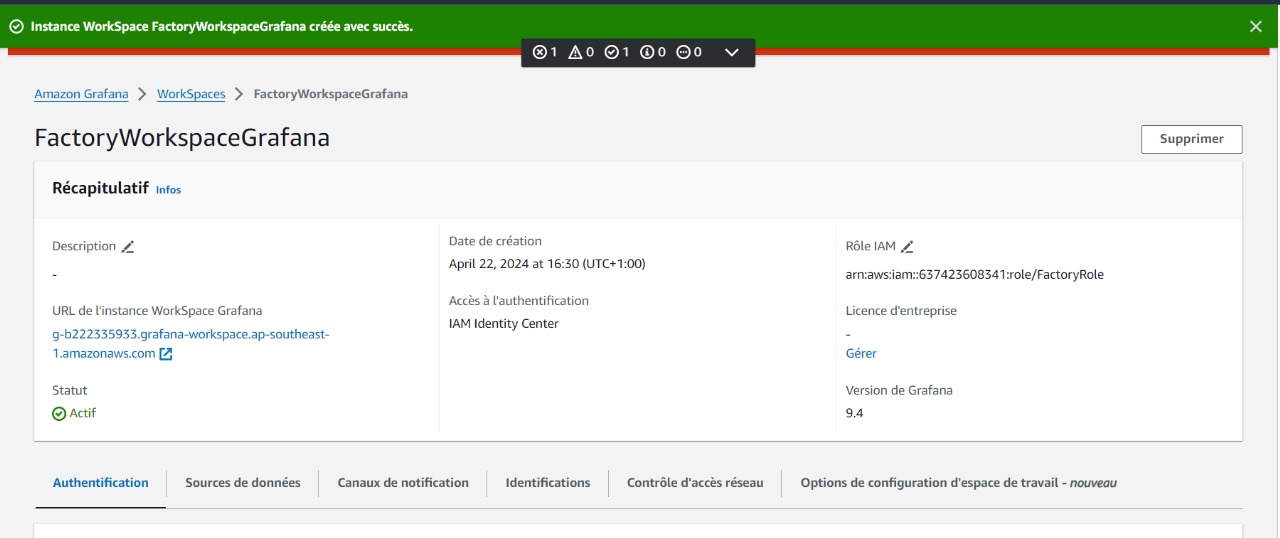
* Model Shader:



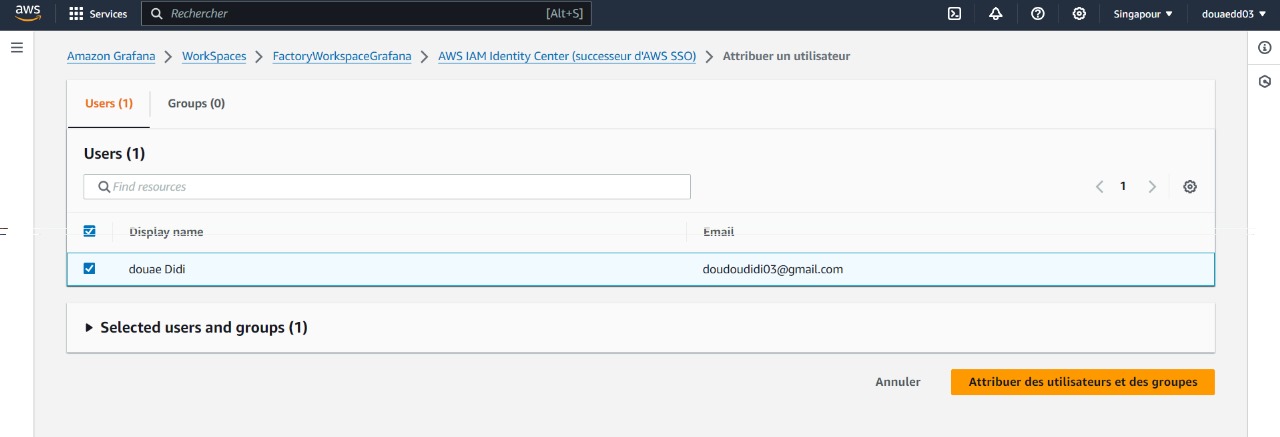
1. Grafana :

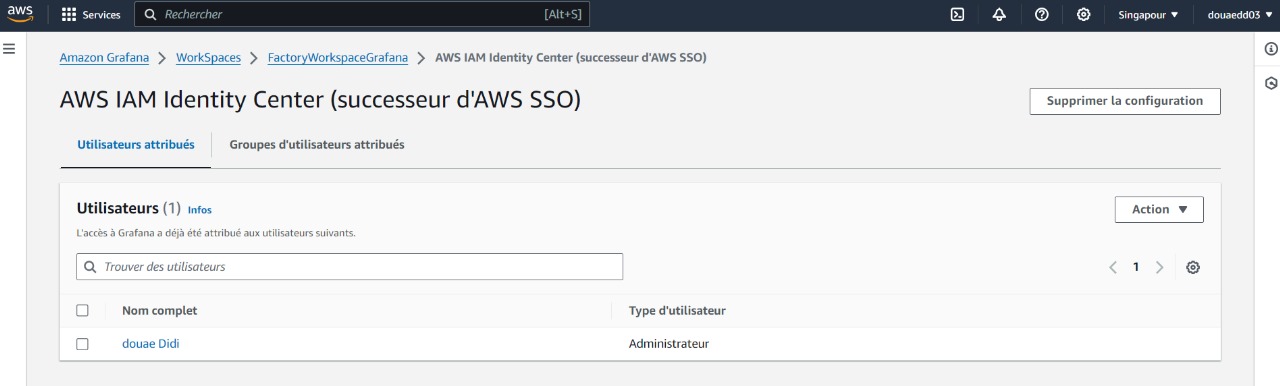
* Creating workspace:





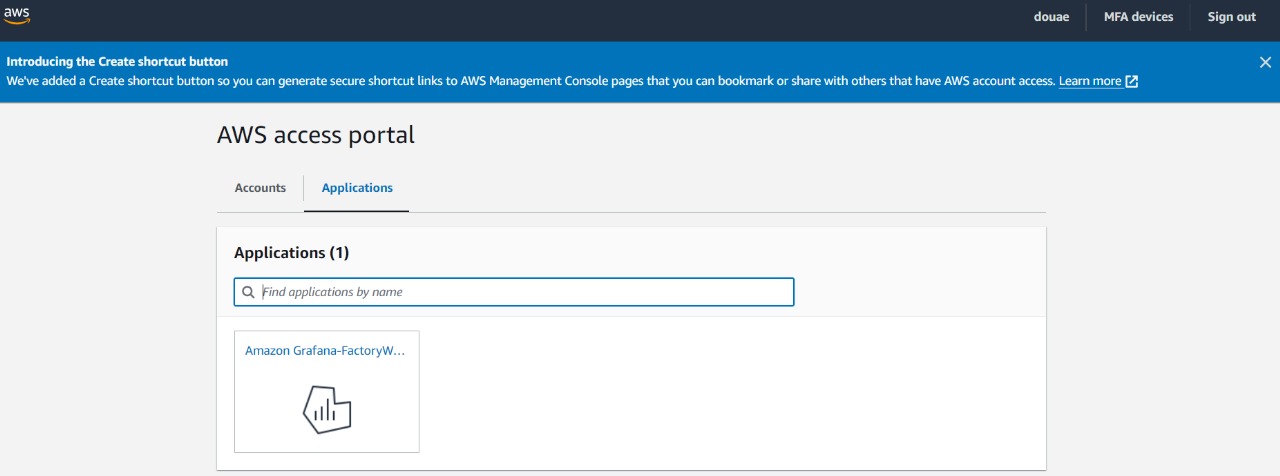
* IAM Identity Center :Add User:

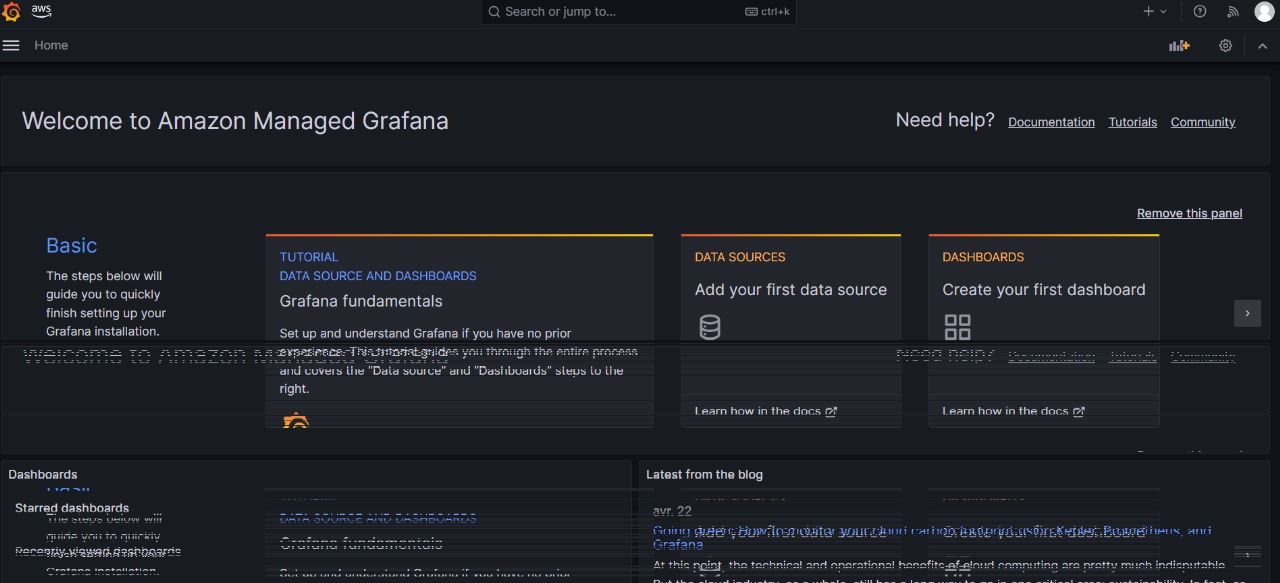




* Login:







* Add Data Source :

**(Partie payante )**

**Partie 3:** ajouter un autre moteur avec autre mesure (température) :

