Service Consumption Report

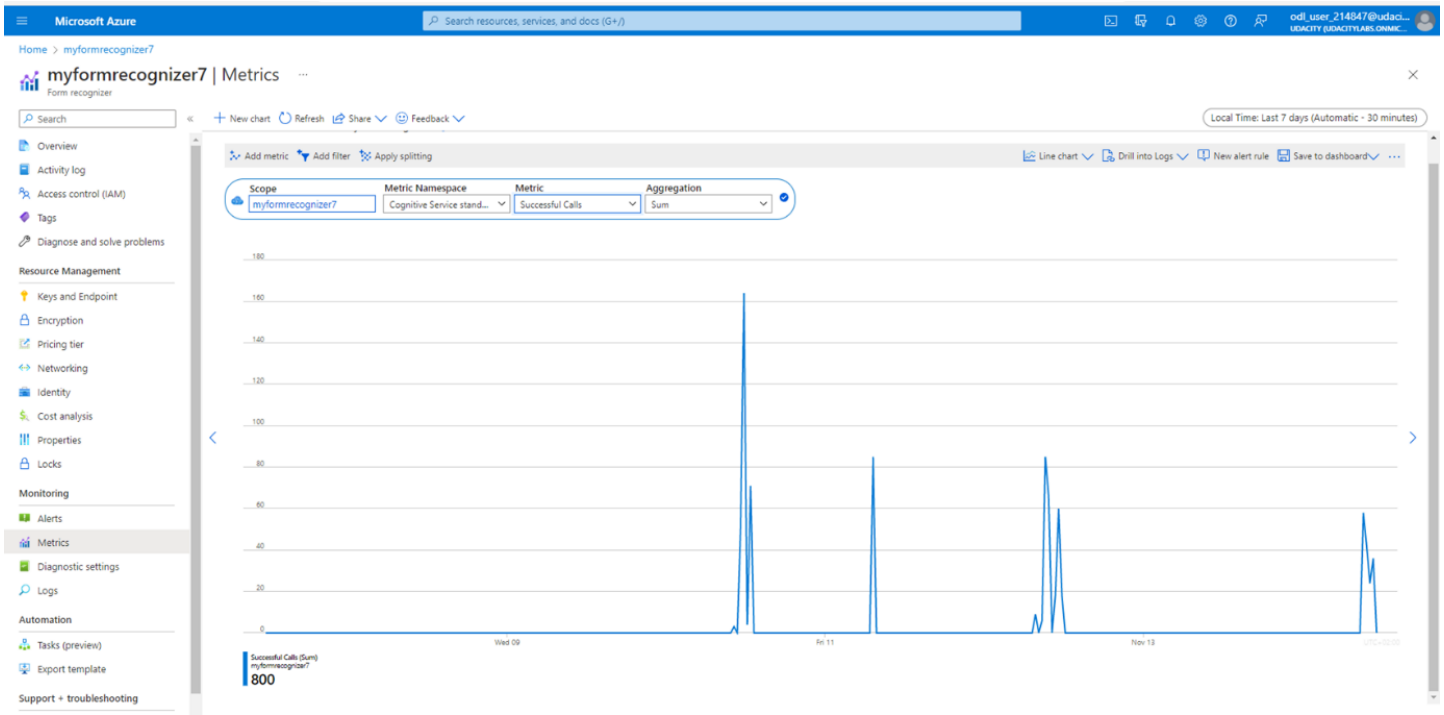
The report aims to provide insight into resource utilization for the Microsoft Azure workspace. The metrics will guide us in understanding how the application uses the resources that have been allocated to it. This will allow use for easier decision-making about resource allocation when deploying the system in the real world.

The report contains service utilization of Azure services:

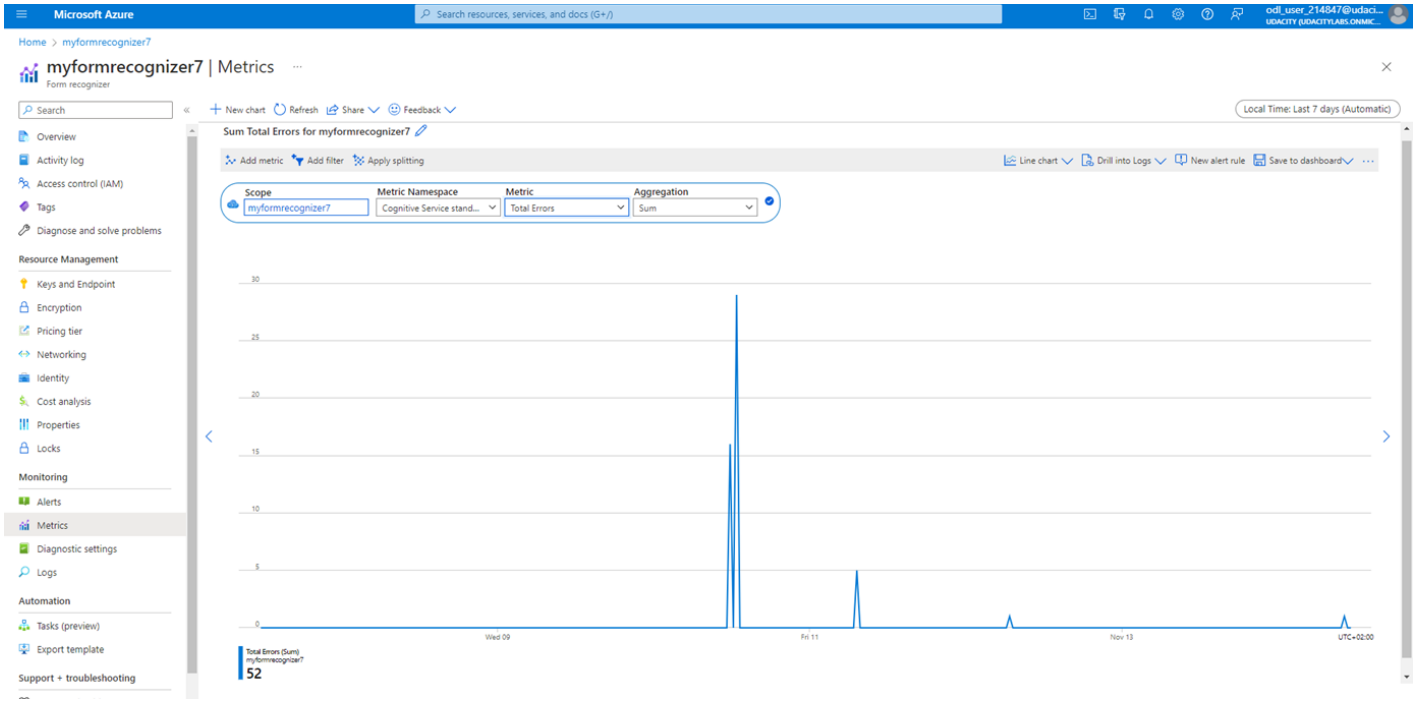
1. Form Recognizer
2. Face API
3. Custom Vision

1. Form Recognizer:

Here you can see the number of successful calls to the service. In total ***800 calls*** were made. This includes both training and inference calls to the service.

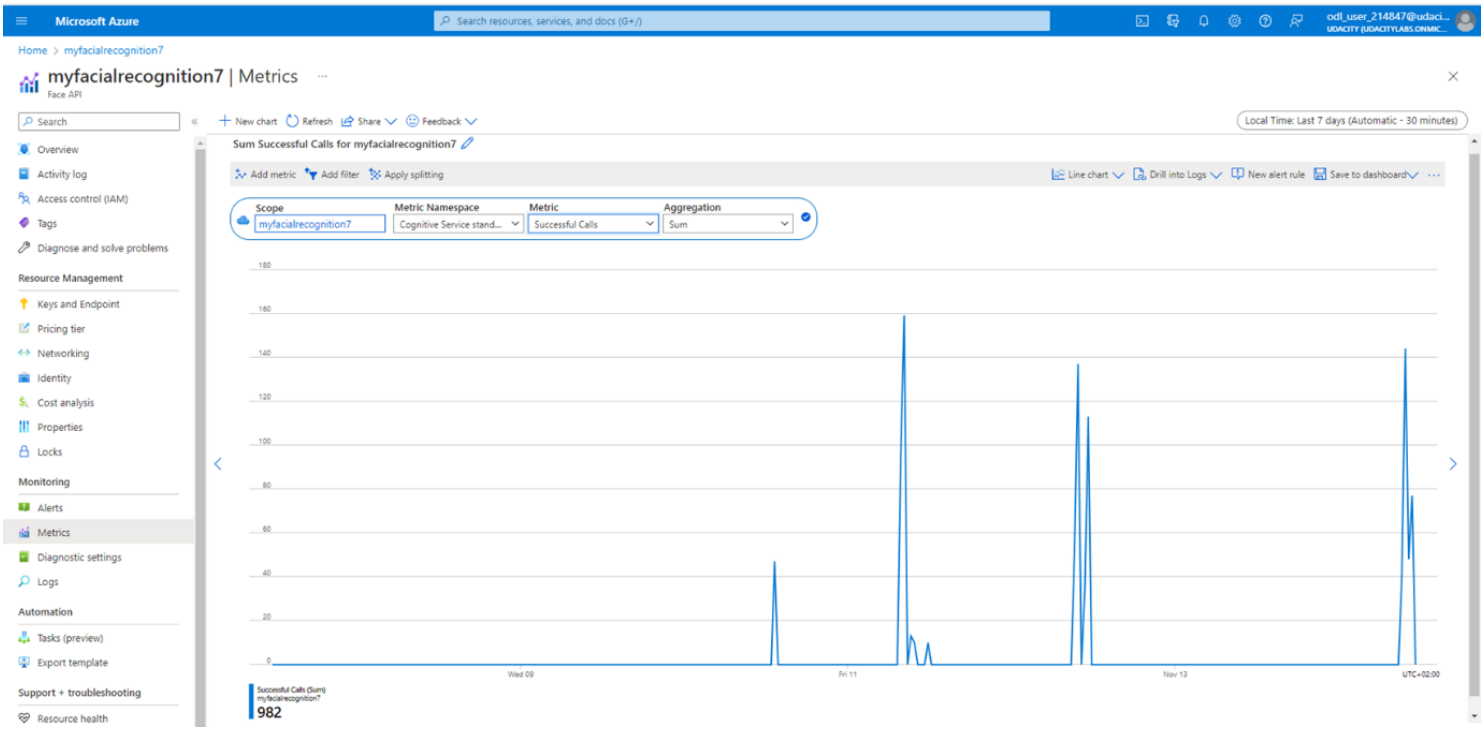


Here we can see the number of unsuccessful calls to the service. It can be seen that ***52 calls*** resulted in errors. The initial spike may be due to the key expiring due to the change in Azure environments, and the key was invalid.

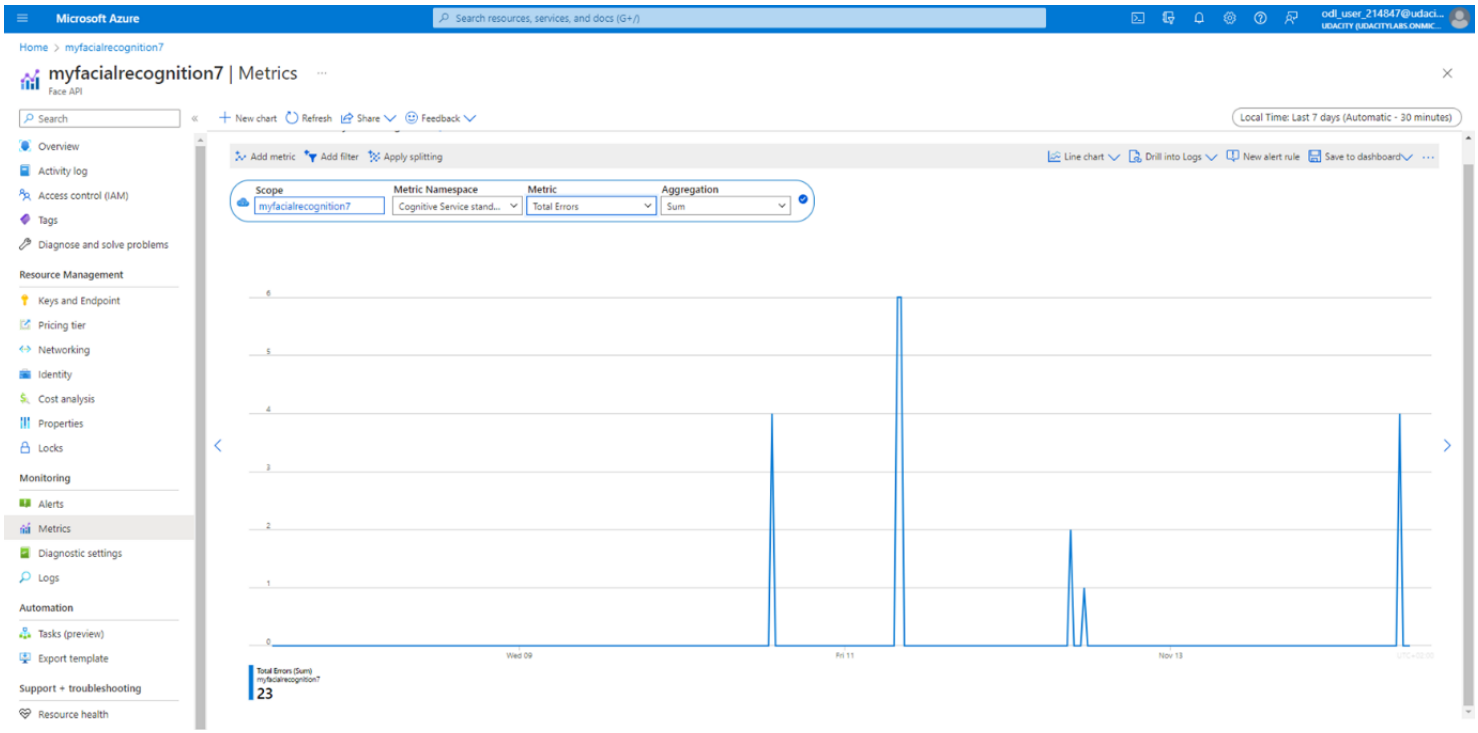


2. Face API

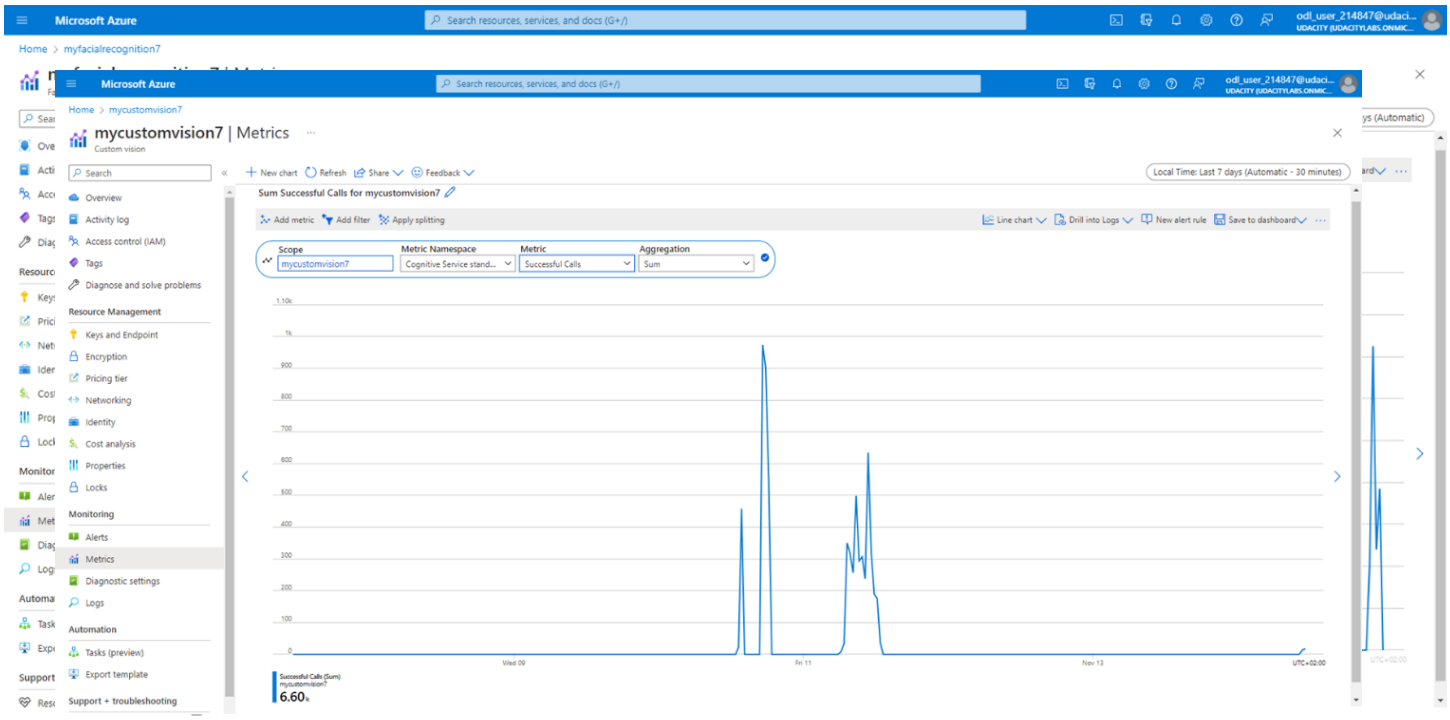
Here the number of successful calls can be viewed. The total number of successful calls are ***982 calls***.



Here the number of errors can be viewed. The number errors amounted to 23 errors in the seven-day period. The initial errors may have been due to the invalid key submission when moving to a new Azure environment. The later errors may be due to exceeding call threshold - which resulted in service denial.



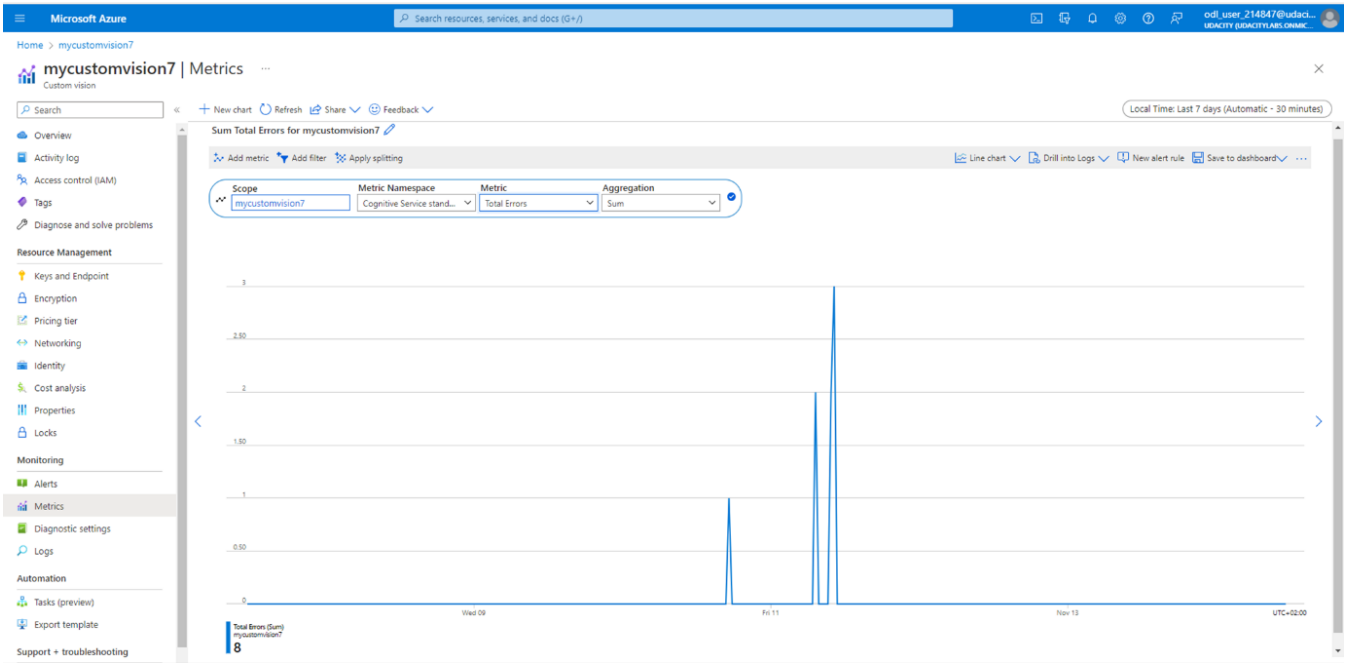
Below we can see that the service extracted facial image data ***990 times***.



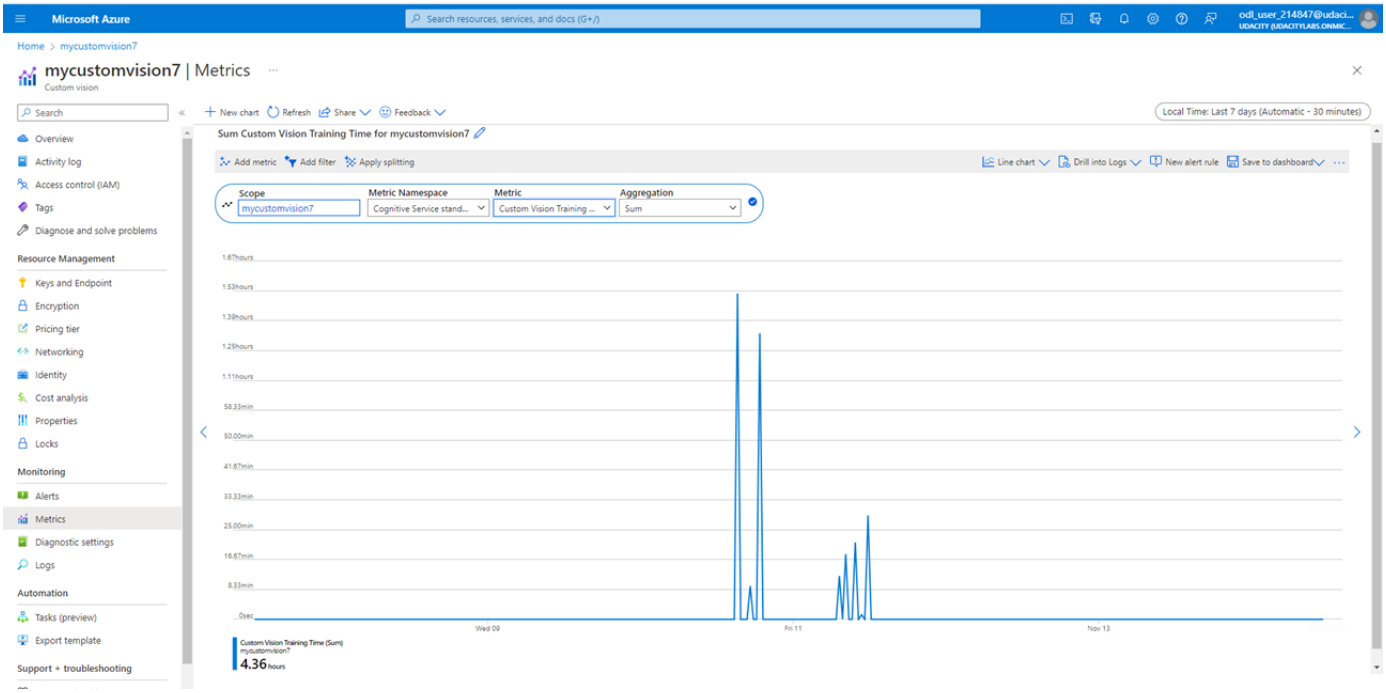
3. Custom Vision

Here the number of successful calls can be seen. The program made 6600 successful calls to the service. This may have been the uploading of images and training of the custom object detection model.

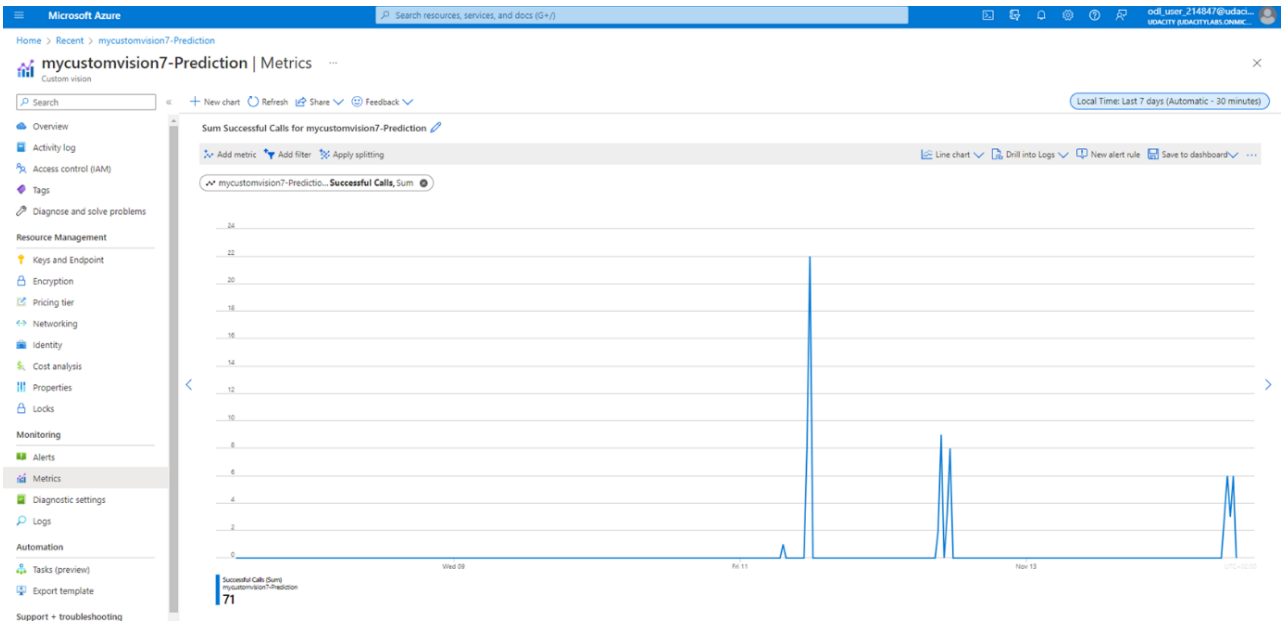
Here the number of errors to the service can be seen. There were only ***8 errors.*** This may be due to constraints on the resource because the free tier was used.



Below, the amount of training time required to train object detection model can be seen. These include all training times for 10 models. The final model was selected for deployment. In total, ***4.36 hours*** were used to train the model.



Moving onto the prediction service of the Custom Vision service, we can see the number of successful prediction calls to the service. In total, ***71 calls*** were made.



Here the number of errors can be seen. There were no errors when making predictions using the Custom Vision service.