EGS: User Guide

Introduction

Elastic GPU Service platform provides a system and workflows for effective resource management of GPUs across one or more kubernetes clusters.

EGS supports two different personas: Admin and User

Admin
Admin is responsible for the installation and administration of EGS platform. EGS provides an Admin portal to perform the Day 0/1/2 operations. EGS also supports YAML (manifests) based admin workflows for these operations so that these workflows can be integrated with CI/CD or MLOps pipelines.

User:
A User (can be a Data Scientist, Researcher or ML engineer) uses EGS User portal to create and manage the life-cycle of GPU provisioning requests for User's Slice Workspace(s).
The GPU provision requests (GPRs) can be created and managed using EGS APIs or YAML/GPR custom resources by User's CI/CD or RAG pipelines or an external system/service or an application service in the cluster as well. EGS User specific UI portal provides deep visualization of the AI workloads and associated GPUs metrics and other data.

EGS Documents

This document describes the EGS User operations related workflows:

- For EGS platform overview please see the documentation on the website
- For Admin guide please see the documentation on website

- For Installation guide please see the documentation on github repo

Table of contents

Get Access to Slice Workspace

A User (or a team) can have one or more Slices (workspaces). Slices can be single cluster scoped or can span across multiple clusters

User works with the EGS Admin to get a Slice created for him/her or team. EGS admin manages the life-cycle of the User Slice workspace. To get access to cluster slice workspace (namespaces) and access EGS UI portal, send an email to the Admin with the following details:

- Name of the workspace (optional)
- Name
- Namespace name
- Email ID

Admin is responsible for creating the Slice workspace with User details. Admin sends Slice workspace cluster's Kubeconfig file and UI access token/portal URL.

Users can then use the Kubeconfig file to access the cluster Slice workspace namespaces.
Users can use the access token to access the EGS User Portal.

Login to the User Portal

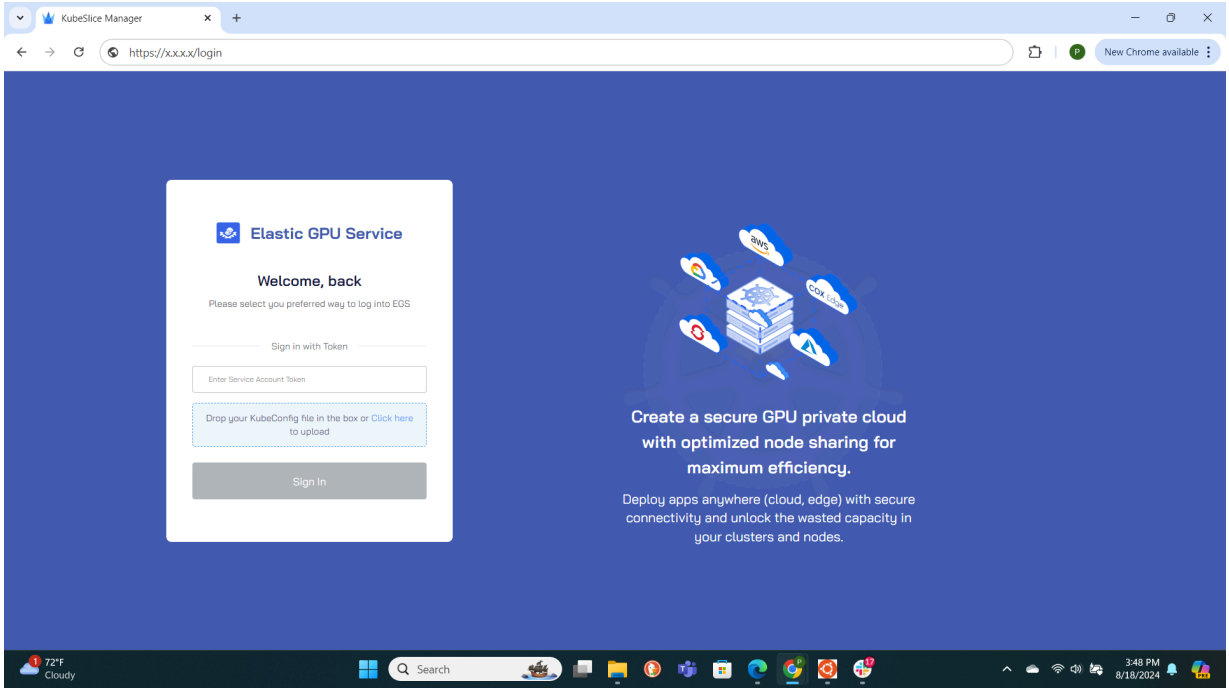
EGS User Portal

Users can access the EGS portal with *Access* token (or with IDP credentials if IDP is enabled on the cluster).

EGS Portal enables workflows to manage the life-cycle of GPU provision requests (GPRs), deep GPU observability for User AI workloads.

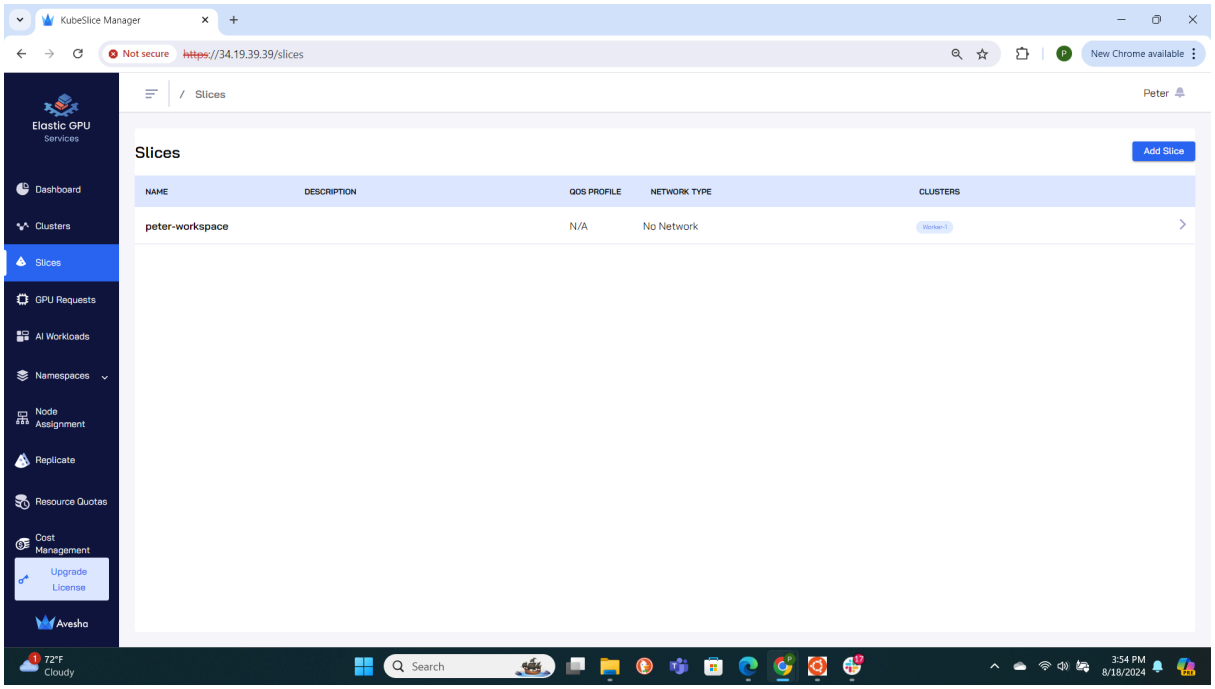
Log in to the EGS portal using the access token using the URL and the token received from the Admin.

📒**Note**: Contact your Admin if you don't have the EGS portal URL or access token.



View Slice Workspace

1. Select **Slices** on the left sidebar.
2. On the **Slices** page, you should see your Slice workspace under **Slices**.



Create GPR

GPU Provision Requests (GPR)

By default, no GPUs will be assigned to User's Slice VPC. To run AI workloads (in the namespaces that are associated with the Slice) that require one or more GPUs, User needs to use the Portal to create a GPU provision request.

Note: GPRs can be created by additional methods using EGS APIs or by applying GPR custom resource YAML to the KubeApi server. These methods can be invoked by the CI/CD or RAG pipelines or external system/services or application services in the cluster.

GPR features:

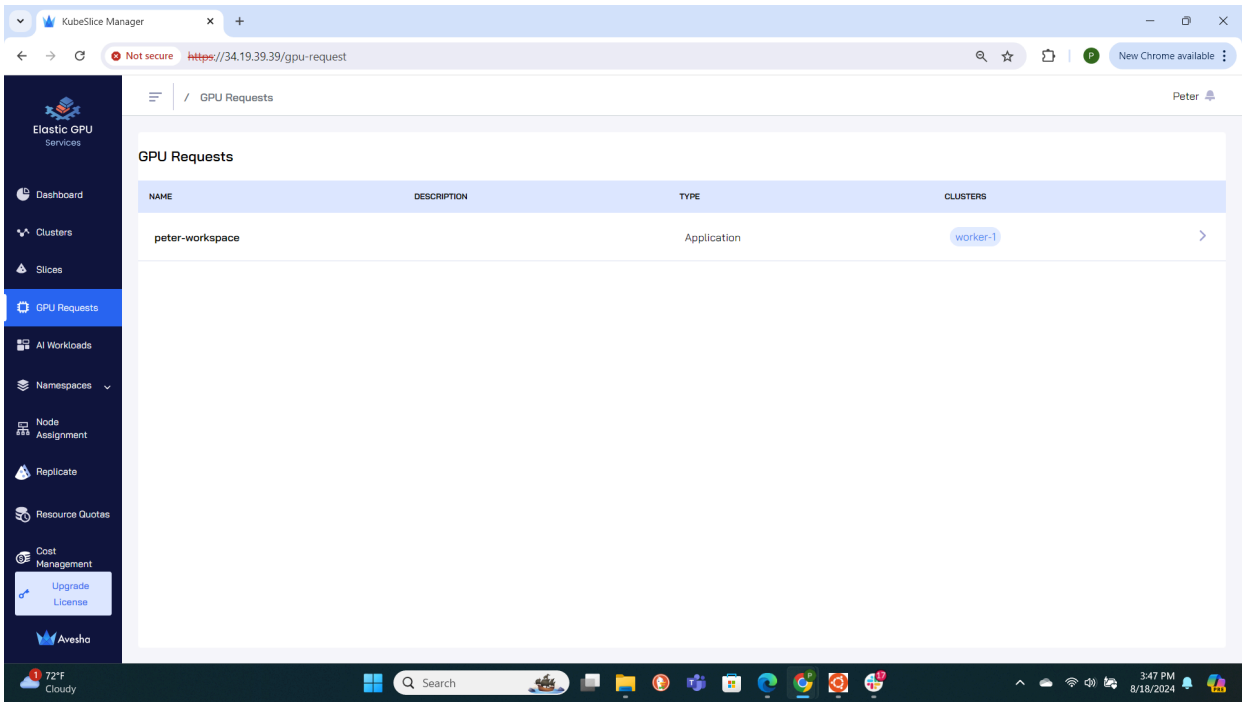
- Users can create one or more GPU provision requests
- Only one GPR will be provisioned in to the Slice at a given time
- GPR has strict entry and exit times for GPU nodes from Slice VPC
- Isolation of GPU nodes per Slice VPC

- Other Slices (or Users) cannot use the GPUs allocated to the User's Slice VPC inadvertently
- Self-service mechanism for GPU provision requests
- Visibility into wait-time for GPUs
- Users can delete/edit GPRs before they are provisioned
- Users can early-release GPR if they no longer need the GPUs in their Slice VPC

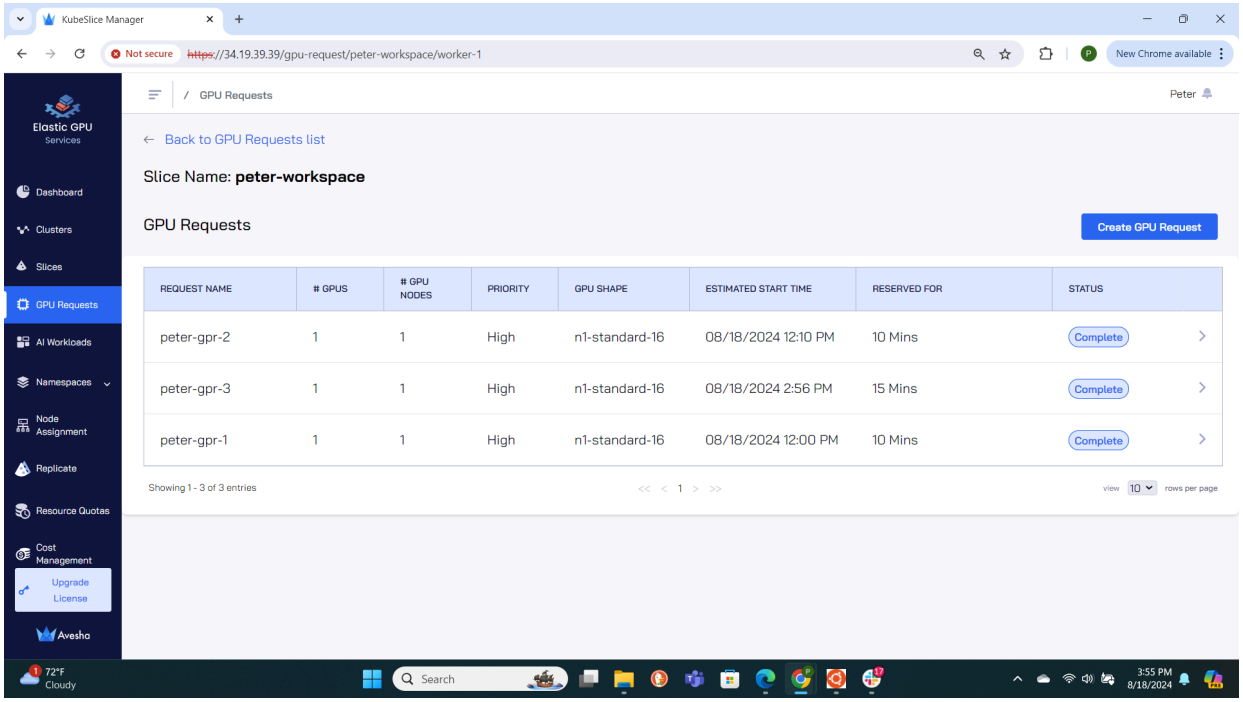
To create GPU requests:.

1. Click **GPU Requests** on the left sidebar.

2. Select the workspace for which you want to create a GPU request.

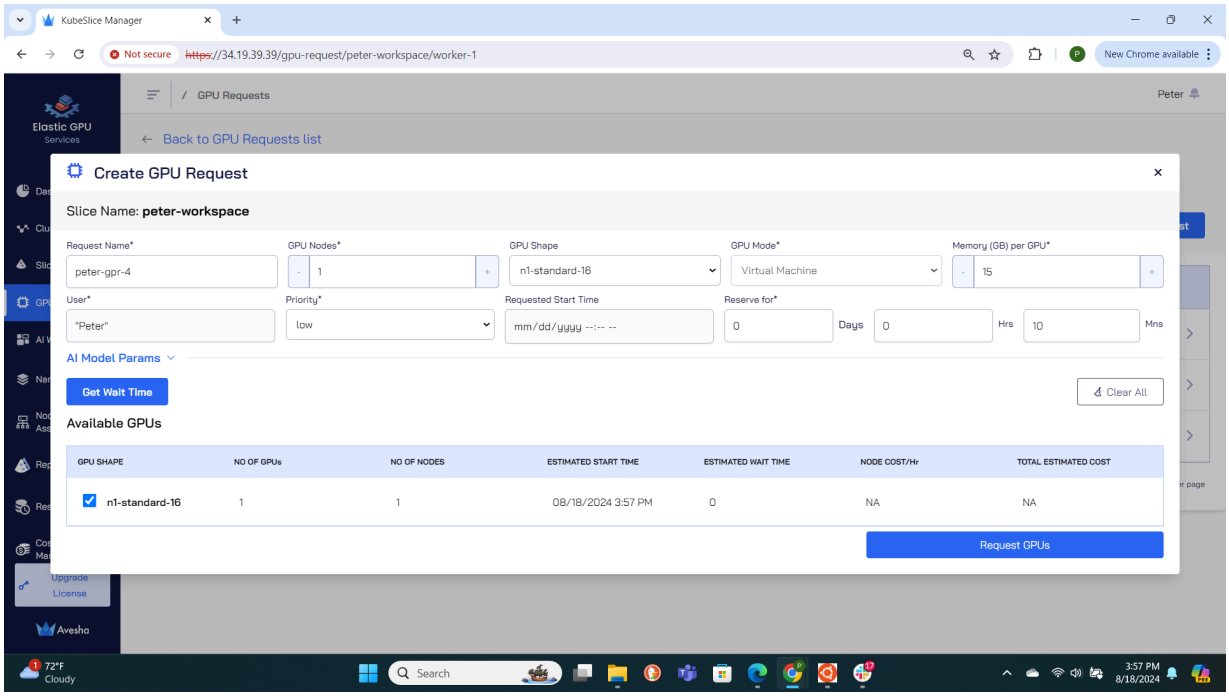


Click **Create GPU Request**.

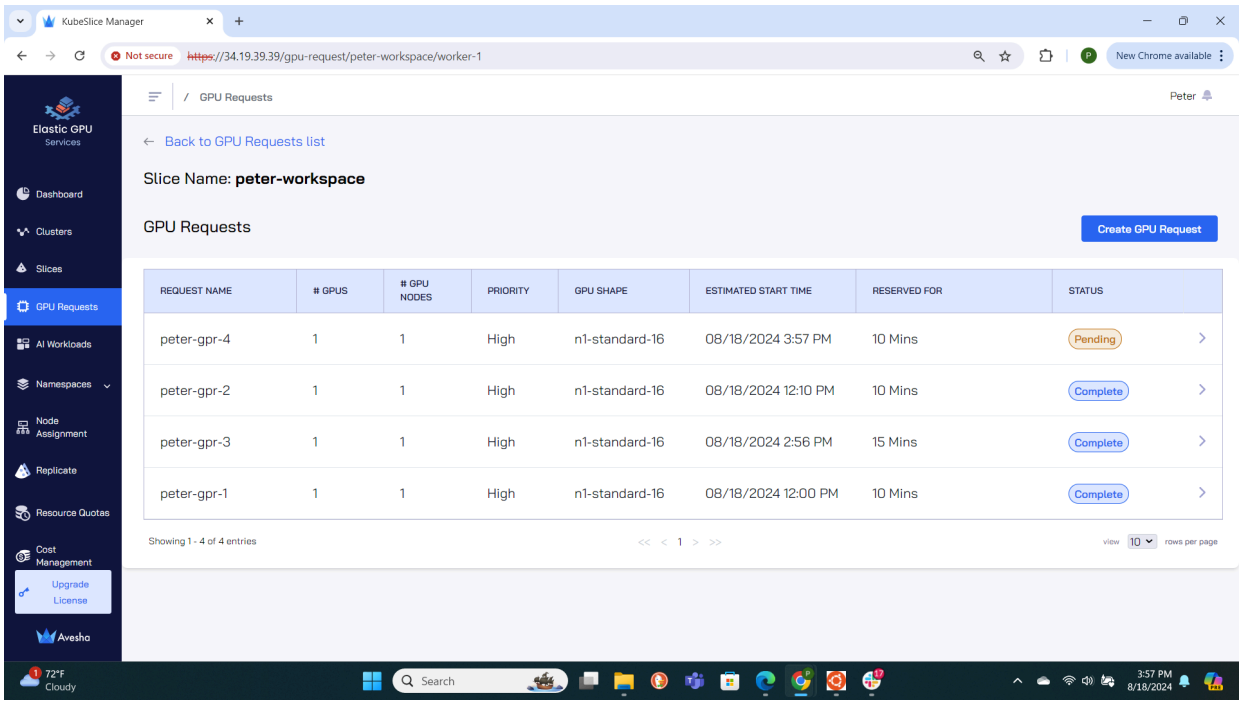


On the Create GPU Request page, add the request details:

1. Enter GPR name and Number of GPU nodes
2. Select GPU shape
3. Select priority
4. Select Reserve for duration
5. Click the **Get Wait Time** button. EGS shows the estimated wait time for the GPU nodes provisioning.
6. Select the GPU in the “Available GPUs” table with acceptable estimated wait time.
7. Click **Request GPUs**.

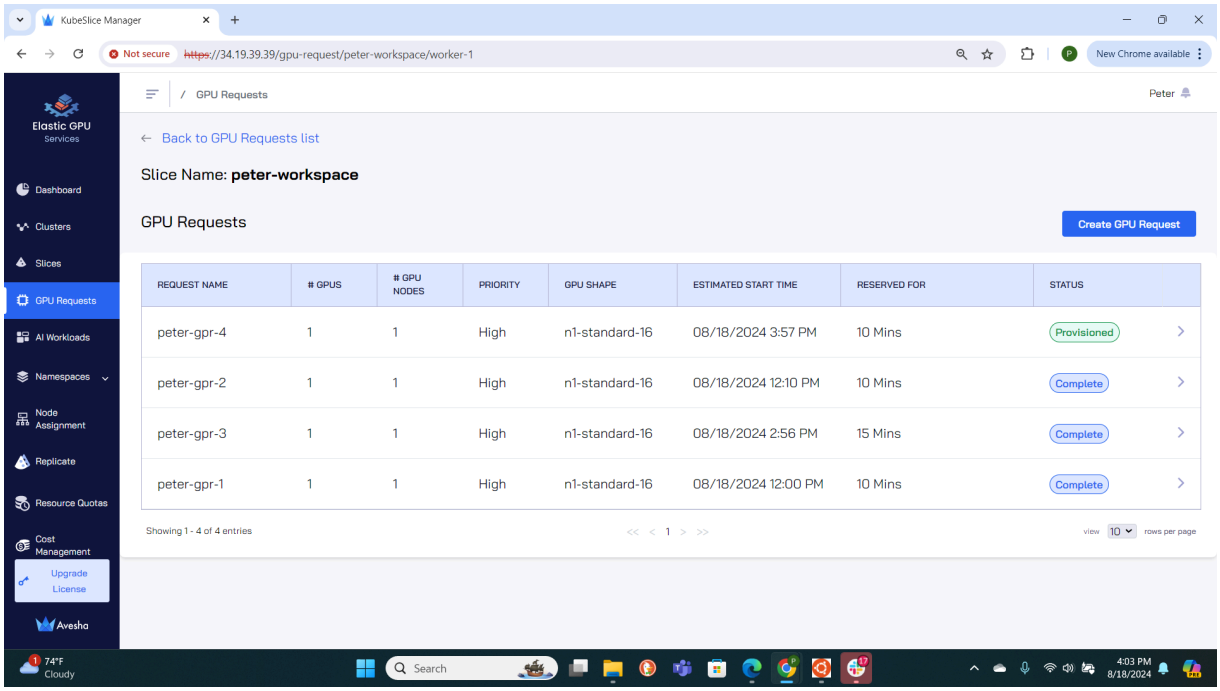


You should see the new GPR with pending or provisioned status in the GPR table now.

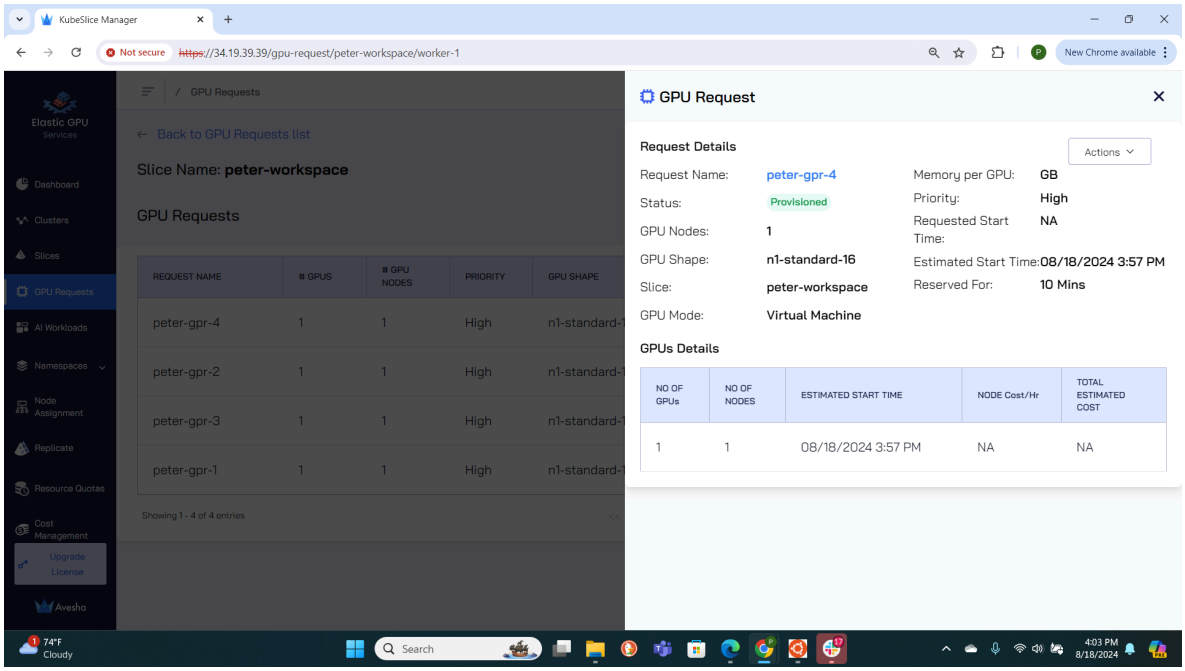


View GPR Queue

Users can manage the GPRs in their Slices. Select the Slice whose GPR queue you want to view.



Click **GPR** to view the request details.



Manage GPR Queue

User can manage the GPRs that are in their Slice(s) GPR queue(s).

The following operations can be performed:

- User can delete a pending GPR
  - This will remove the GPR from the queue
- User can early-release a provisioned GPR
  - This will end the GPR early (early exit of GPU nodes)
- User can edit a pending GPR (available in next release)
- User can extend a GPR with a small grace period (available in next release)

Early release a GPR

User can early-release a provisioned GPR

If for some reason User wants to release the GPU nodes associated with the Slice, User can early-release the GPR.

Select the GPR and open *Actions* menu to see the early-release option

📒**Note**: once the GPR is early-released, the GPU nodes will no longer be available for any AI workloads running in Slice workspace. Any running Workloads (pods/etc.) that were using GPUs and running on the node will go into pending state.

Deploy AI Workloads

User can access the Cluster namespaces using the Slice workspace KubeConfig YAML file received from the Admin. User can deploy the AI workloads only in these namespaces. Note: User Slice VPC is isolated from other users or Slices. The GPR provisioned GPU nodes are available for the User for the duration of the GPR.

📒**Note**: Contact your Admin if you don't have Kubeconfig YAML for the cluster.

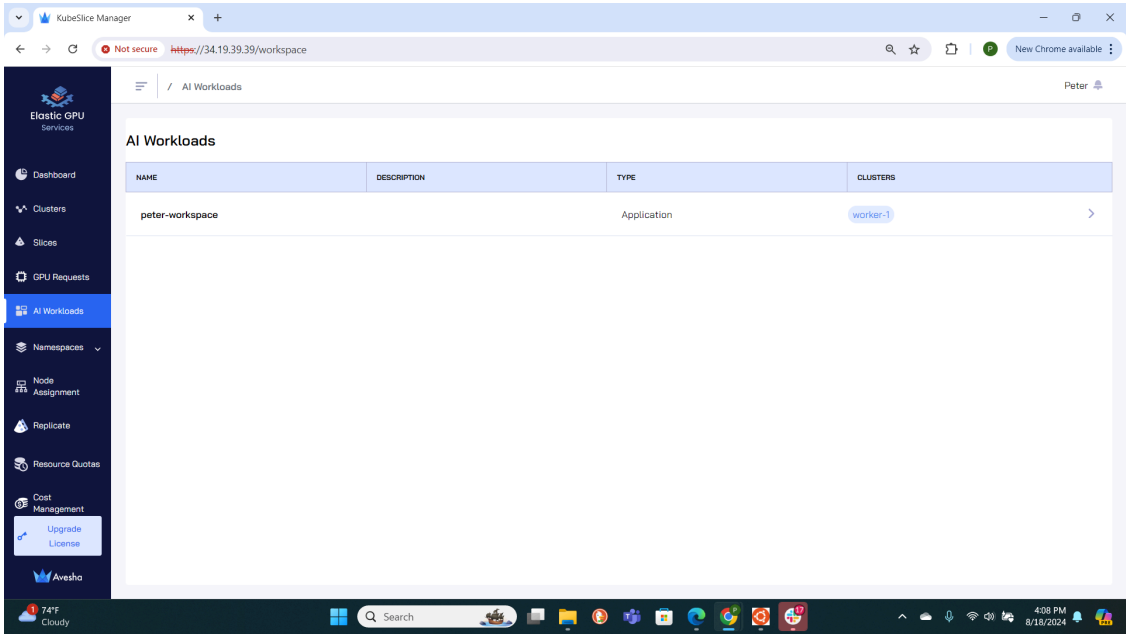
View AI Workloads

Users can view AI workloads and associated GPU details that are running in their slice namespaces (workspaces).

EGS provides highly granular visualization for every AI workload and associated GPUs:

- AI workloads lists model, model configuration, and infrastructure committed for the workload (LLM training or fine-tuning job).
- Visibility into high power usage GPU, high temperature GPU
    - Generates alerts on high power/utilization levels
- Visibility into GPU metrics - dashboards for Users AI workloads parameters/GPU metrics

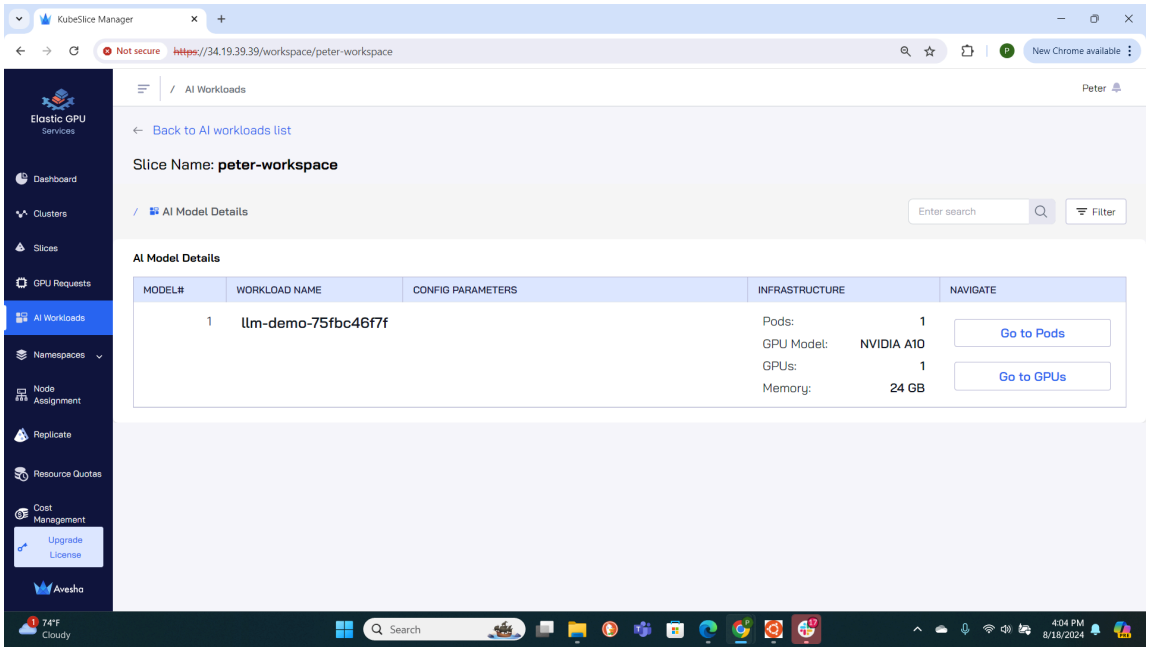
1. Select **AI Workloads** from the left sidebar.
2. Select **Slice** to see the AI workloads for the slice.



Model Details

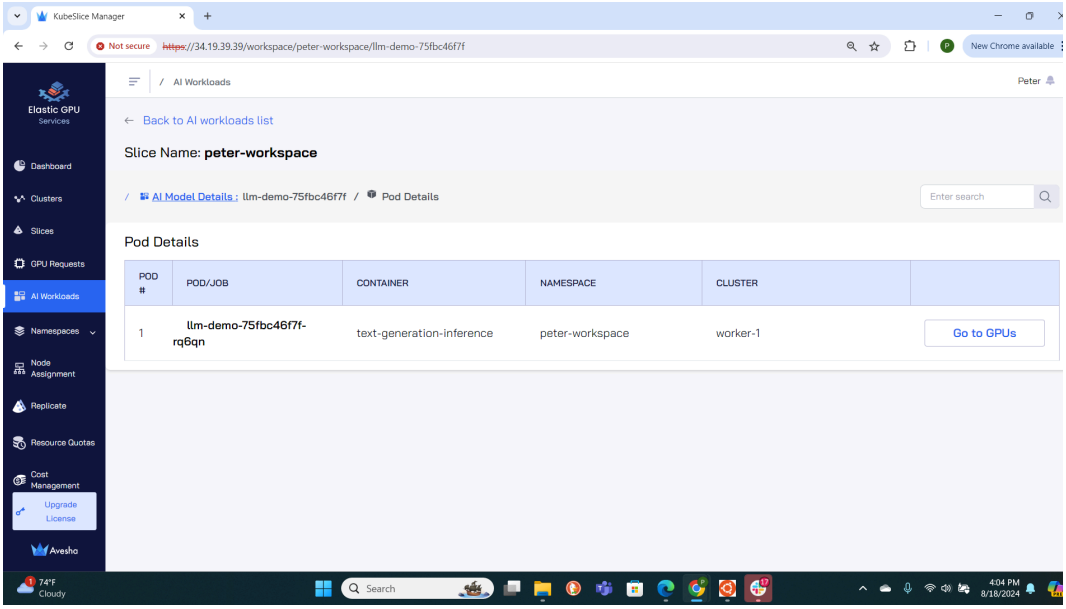
Select **User Slice** to see AI workloads for User workspace and it:

- Shows the model details, GPU infrastructure committed to the workload.
- Shows model summary - high power GPU, high temp GPU and Average Utilization values.



View Pods

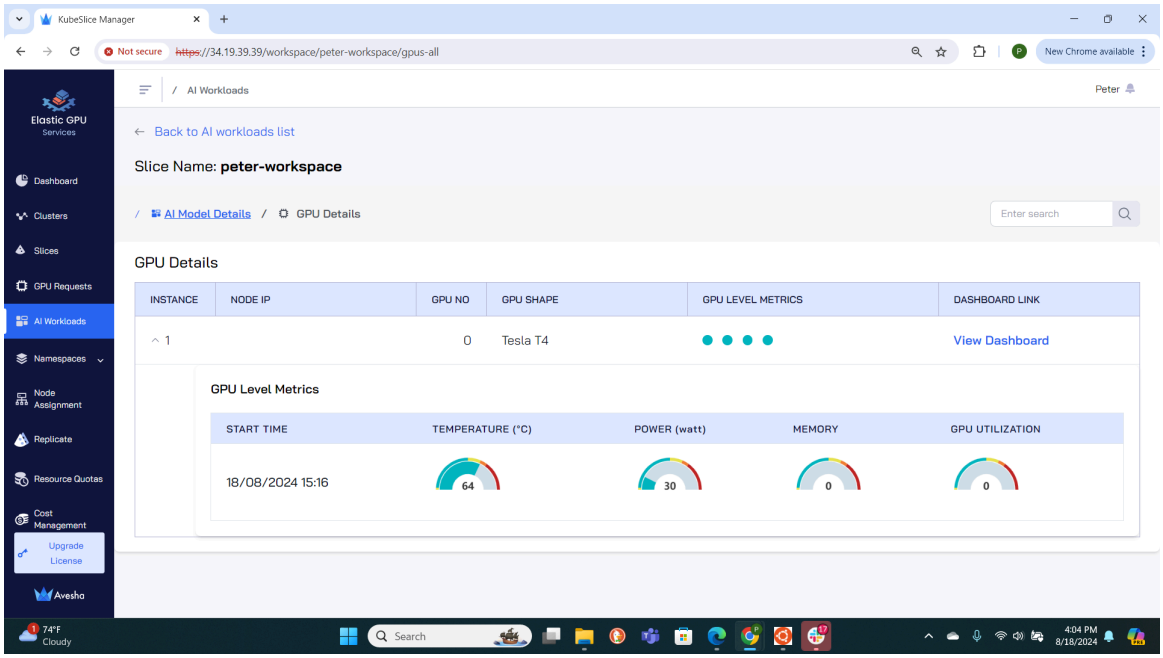
- Click Go to Pods to see the pods running with GPUs in the workspace.



View GPUs

From the **AI Model Details** page, click **Go to GPUs** to see the GPU table page. The GPU table:

- Shows sorted list of GPUs with high power, temperature GPUs at the top for quick access.
- Shows the hotspot GPUs.
- Click **View Dashboard** to view the time-series data for the selected GPU device.



Alerts and Events

User can view events related to the Slice workspace, namespace and GPRs and detailed information about the event.

Click on the Bell icon next to the User name on any page to access the Events table. Events are Kubernetes events. Events can be searched for GPR provisioning and progress and other details.

EGS generates various events during the life-cycle of the GPR - created, provisioned, 25%, 50%, 75% of duration marks, during exit and GPR complete.

Users can get a slack channel provisioned for the User (with Admin's help) to see the events on a User specific slack channel.