EGS: Admin 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 Admin operations related workflows:

- For EGS platform overview please see the documentation on the website
- For User guide please see the documentation on website
- For Installation guide please see the documentation on github repo

Table of contents

Installation

EGS provides a combination of Helm charts, CLI and shell scripts for easy installation procedure.

- git clone the `egs-installation` repo :https://github.com/kubeslice-ent/egs-installation

- Follow the Installation steps specified in the EGS Installation Guide.

📒 **Note**: Installation scripts create a default project workspace and register worker cluster(s).

Admin Access Token

Admins can access the EGS UI portal using two different methods:

1. Using Access Token
2. Using IDP Access Token (when cluster and EGS is enabled with Idp integration)

By default, an admin access token will be created in the system as part of the installation.

Run the following script (from the installation guide) to get the Admin access token:

```
Unset
##%./egs-get-admin-access-token.sh -h

or use the following command

%kubectl get secret kubeslice-rbac-rw-admin -o jsonpath="{.data.token}" -n
kubeslice-avesha | base64 --decode
```

Note down the admin access token to use it on the Admin Portal .

Access EGS UI Portal

```
Unset
%/home/user/egs-installation$ kubectl get svc -n kubeslice-controller
NAME                                                    TYPE
CLUSTER-IP    EXTERNAL-IP   PORT(S)         AGE
gpr-manager                                             ClusterIP
10.7.44.212   <none>        8088/TCP        46h
kubeslice-api-gw                                        ClusterIP
10.7.38.173   <none>        8080/TCP        46h
kubeslice-controller-controller-manager-metrics-service   ClusterIP
10.7.46.137   <none>        8443/TCP        46h
kubeslice-controller-release-prometheus-service         ClusterIP
10.7.43.250   <none>        9090/TCP        46h
kubeslice-controller-webhook-service                    ClusterIP
10.7.40.229   <none>        443/TCP         46h
kubeslice-ui                                            ClusterIP
10.7.42.178   <none>        80/TCP          46h
kubeslice-ui-proxy                                      LoadBalancer
10.7.45.163   34.19.39.39   443:31322/TCP   46h
kubeslice-ui-v2                                         ClusterIP
10.7.39.69    <none>        80/TCP          46h
```

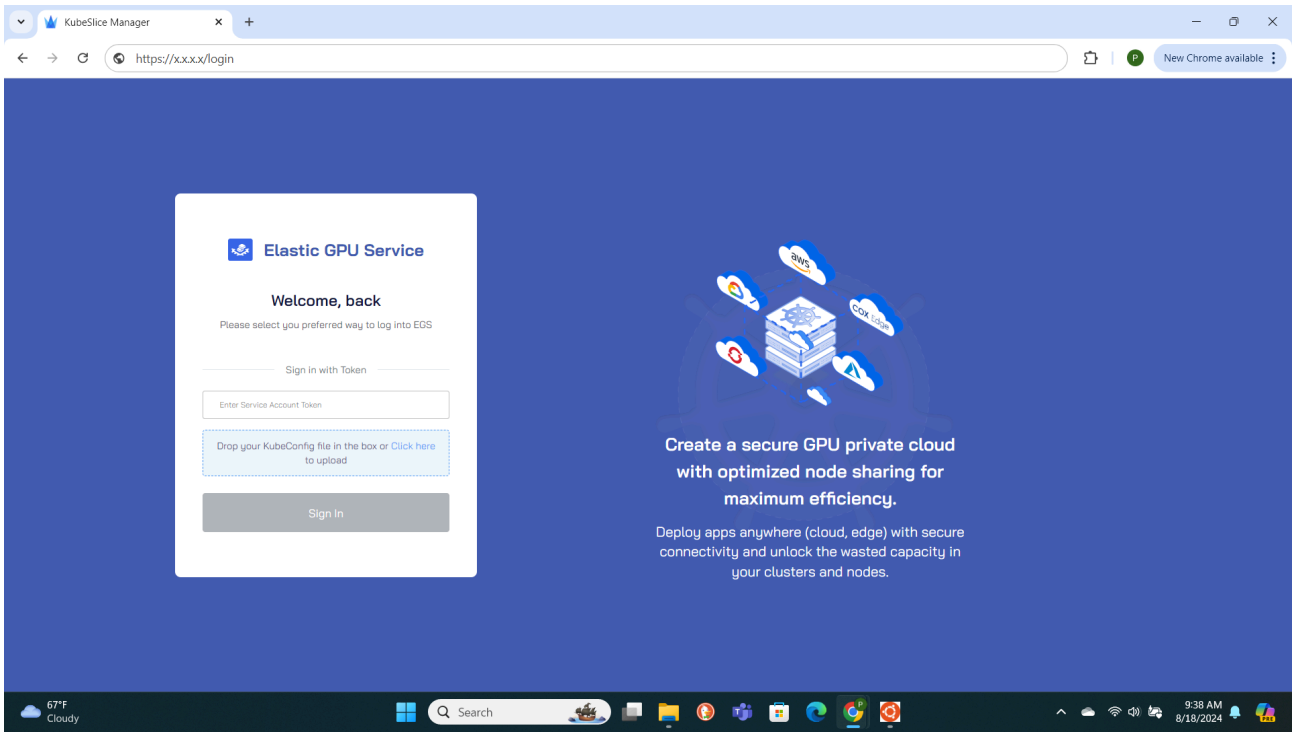
- Note down the LoadBalancer external IP for the kubeslice-ui-proxy pod. This IP will be used to access the EGS Admin/User portals. In the above example: 34.19.39.39 is external IP.

UI Portal: **https://<ui-proxy-ip>**

Login to the Admin Portal

Login to the EGS UI portal with the URL from the previous step.

Use the Admin Access token to log in to the Admin Portal.



Create User Slice (Workspace) Workflow

Admin is responsible for setting up a Slice workspace for a User (or a team).

Ensure that you have User name, namespace, email ID before performing the following steps.

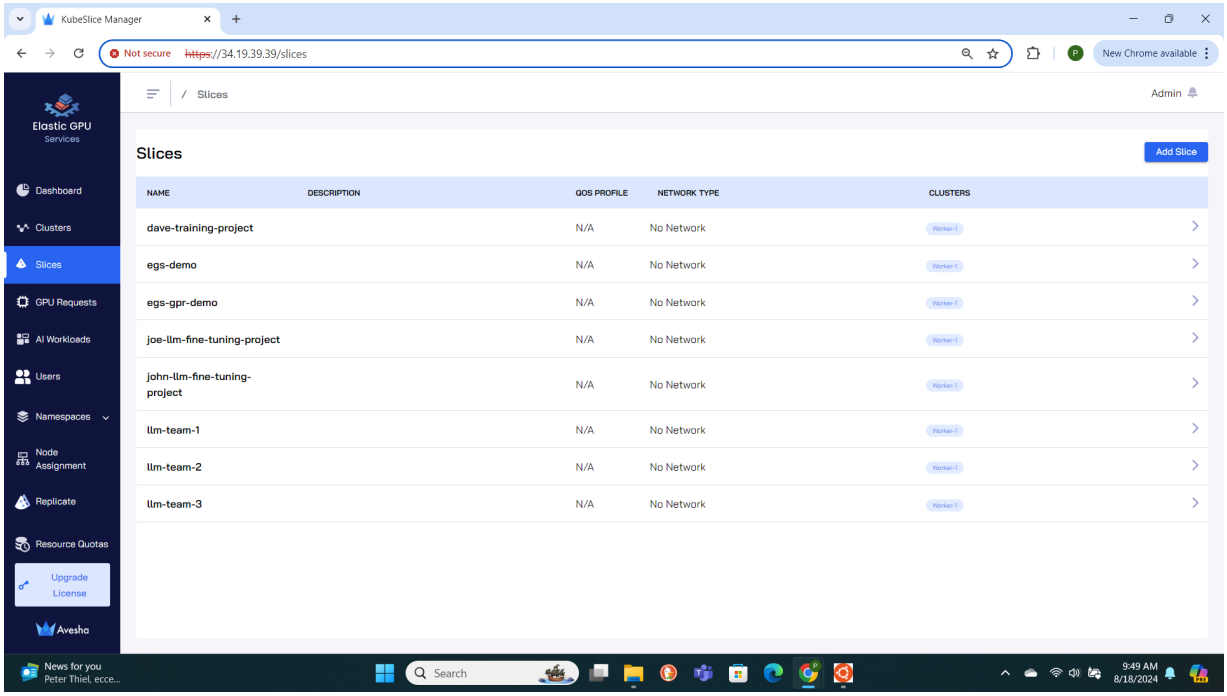
Perform the steps below to set up a User Slice (workspace):

1. Create a Slice for the User (with team name or user name)
2. Assign a namespace to the User Slice (workspace).
3. Add User to the Slice workspace.
4. Provision Slice RBAC for User.
    - Add role and role bindings for the Slice namespace.
5. Get UI Portal Access token for the User .
    - User can access the UI portal using the token.
6. Download KubeConfig for the Slice (from where?).
    - Kubeconfig will have SA token to access the User namespace
7. Send the UI portal Access token to the user.
8. Send the KubeConfig to the user.
    - User can access the namespace on the cluster.
    - User RBAC scope is associated namespace.

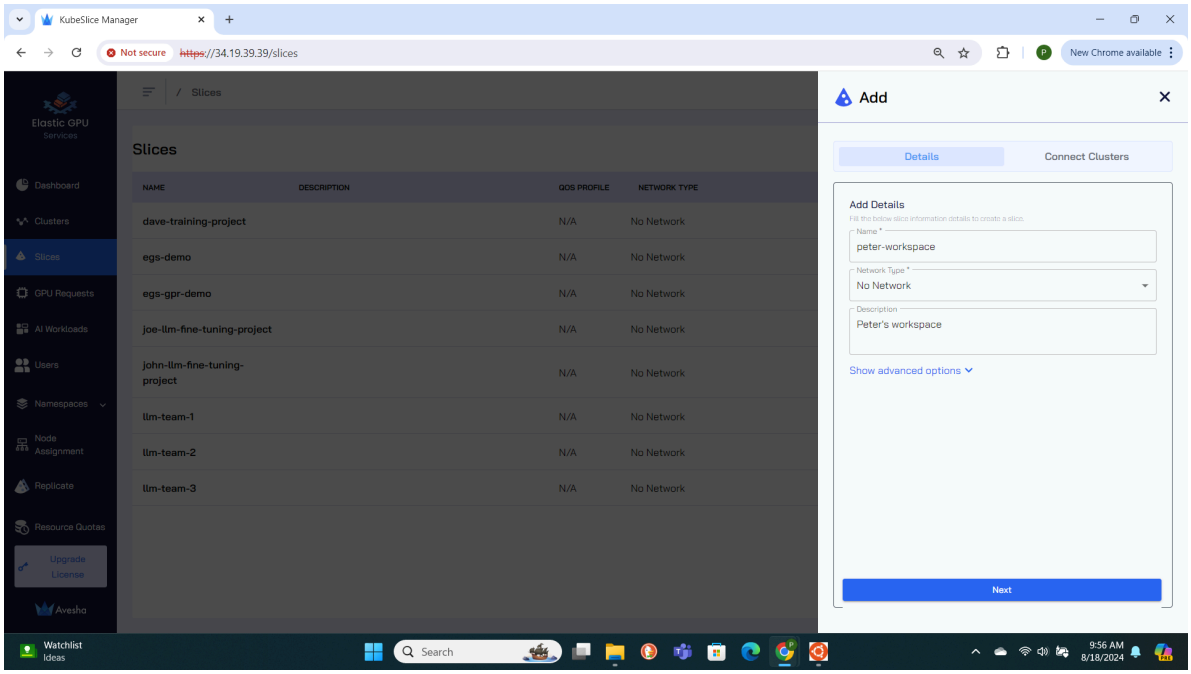
Create Slice for User

Perform the following steps to create a Slice.

Select Slices on the left panel.

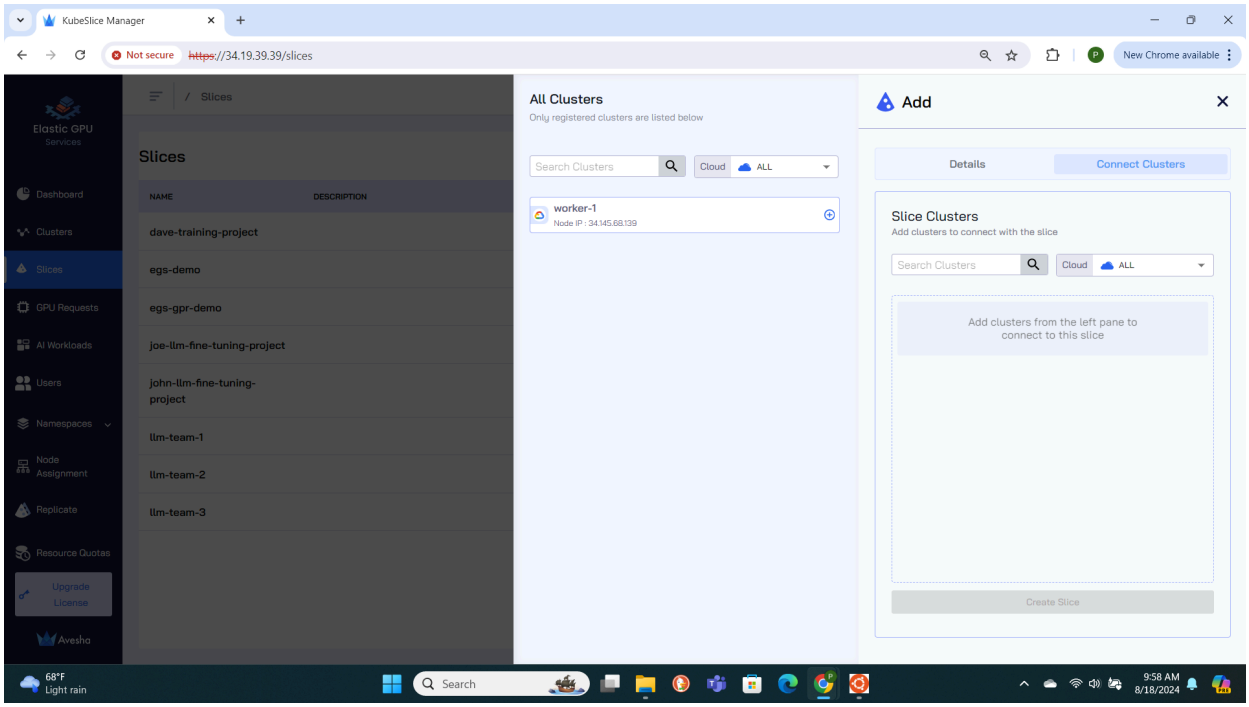


Click **Add Slice.**



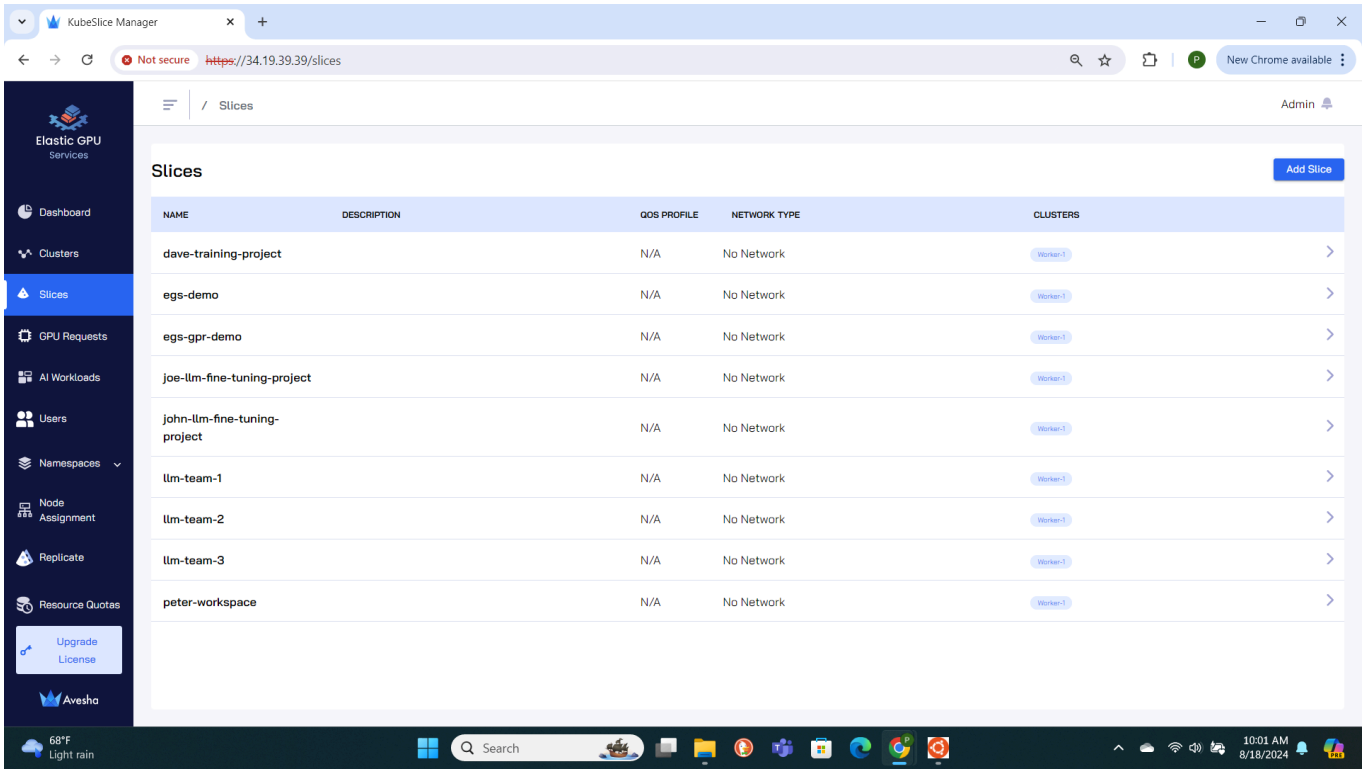
Add slice details:

1. Add **User name**. (example peter-workspace)
2. Select **No-Network** for **Network Type.**
3. Click **Next.**



4. **Add Cluster to the Slice**
    a. Click the + sign to add worker-1 cluster to the User Slice.
    b. **Click Create Slice.**

You should see the newly created slice in the Slices list.



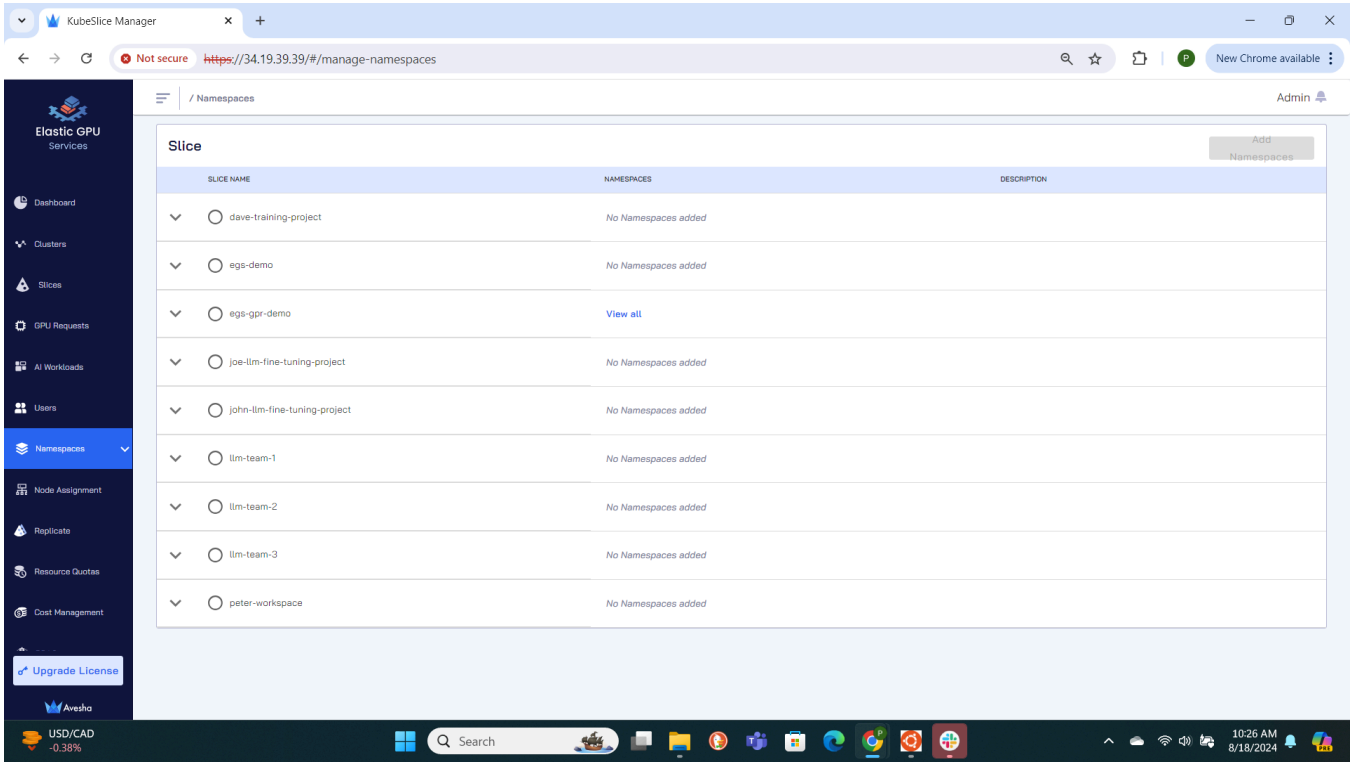
Assign a Namespace to the User Slice

If the user has supplied the namespace for the workspace, use that namespace to associate with the User's Slice. Otherwise, create a new namespace for the user.

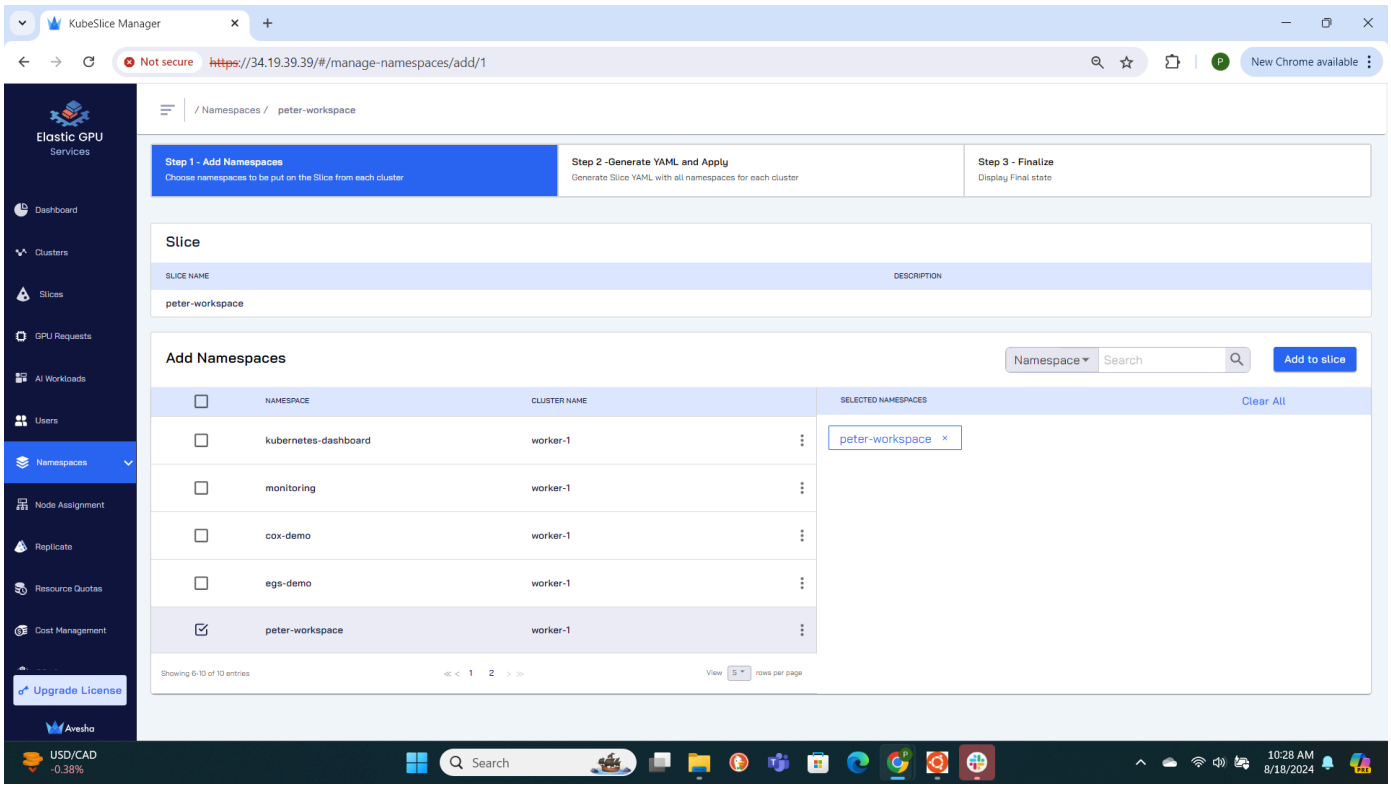
```
Unset
/home/user/egs-installation$ kubectl create ns peter-workspace
namespace/peter-workspace created

/home/user/egs-installation$ k get ns peter-workspace
NAME              STATUS   AGE
peter-workspace   Active   8s
```

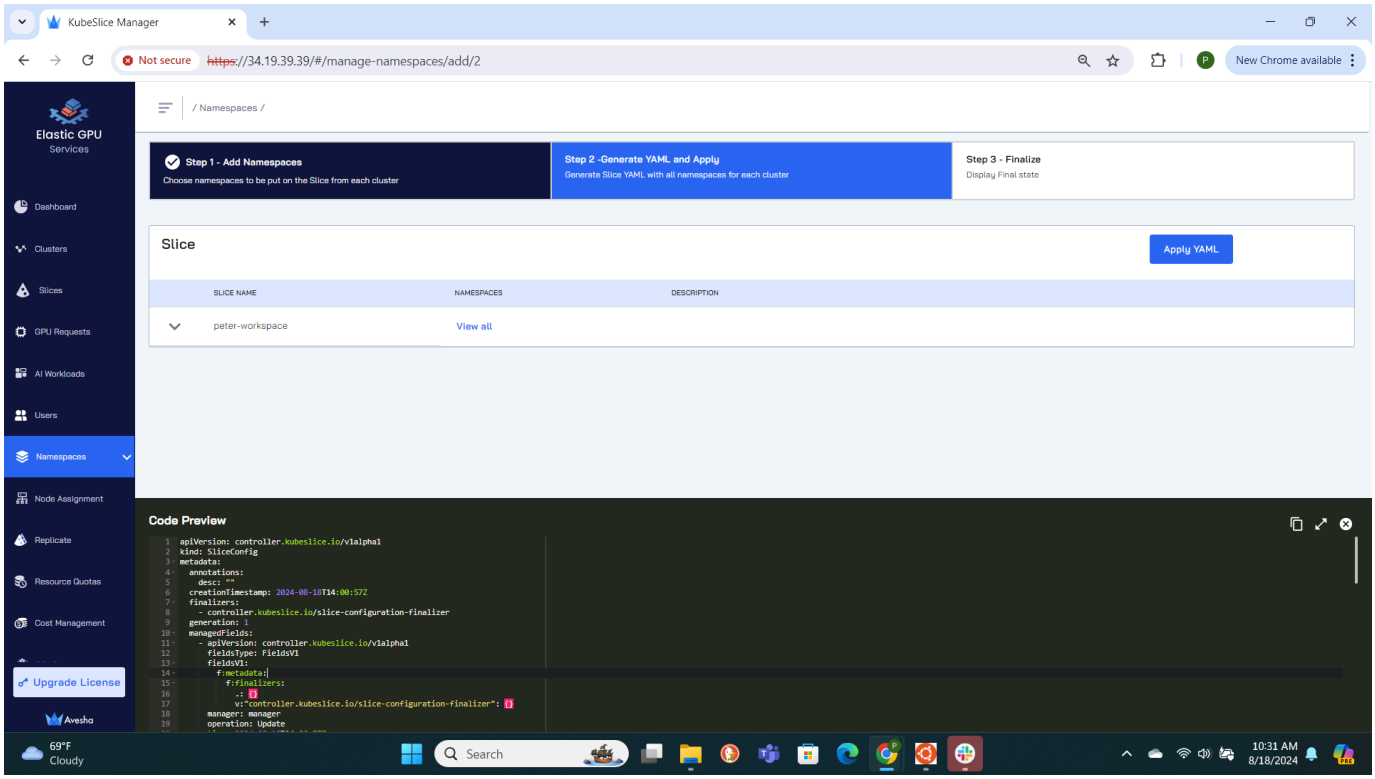
Select Namespaces on the Left Panel



1. Select the **User Slice**.
2. Click **Add Namespaces.**
3. Select the user-workspace namespace.
4. Click **Add to Slice.**



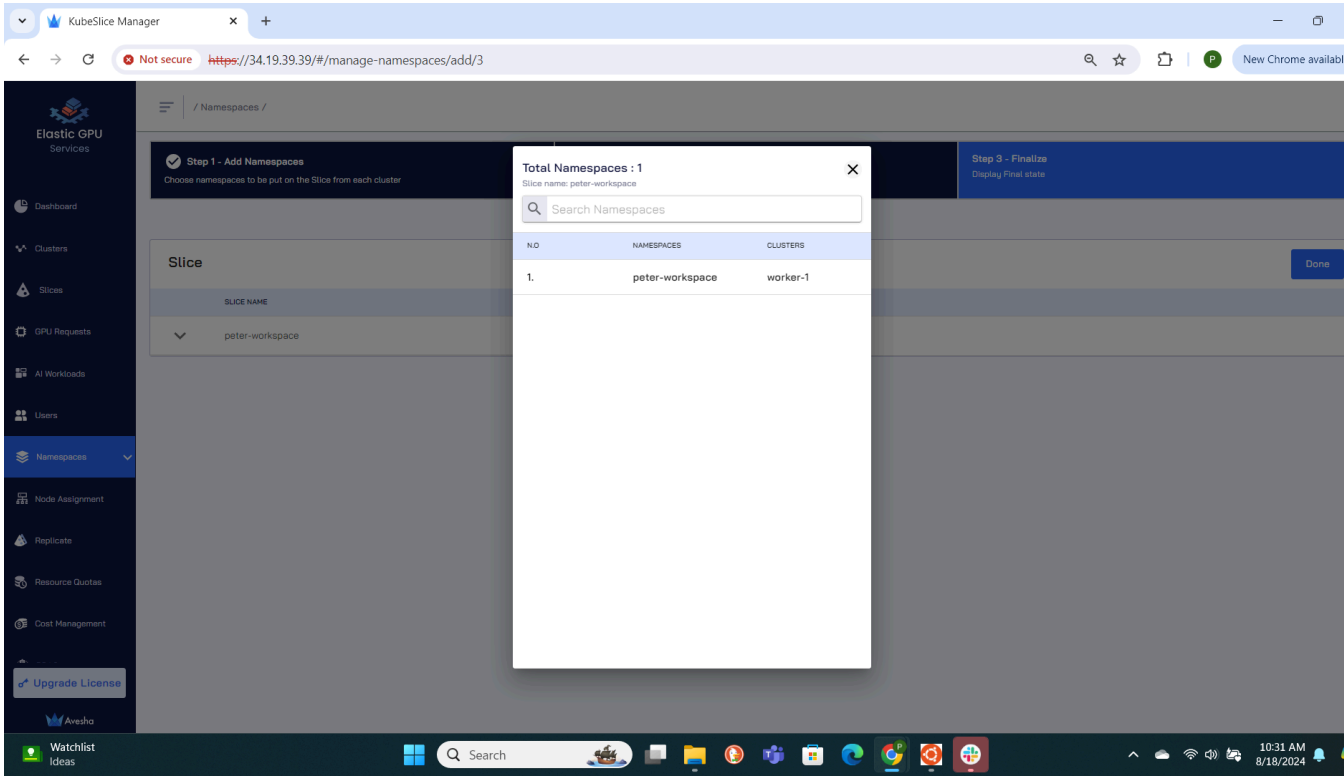
1. Go to the Step 2, Generate YAML and apply tab and click **Apply YAML**



View Namespaces

1. Select the **Slice** that you want to view.

2. Click **view_all** on the Slice bar to see the namespaces associated with User Slice.

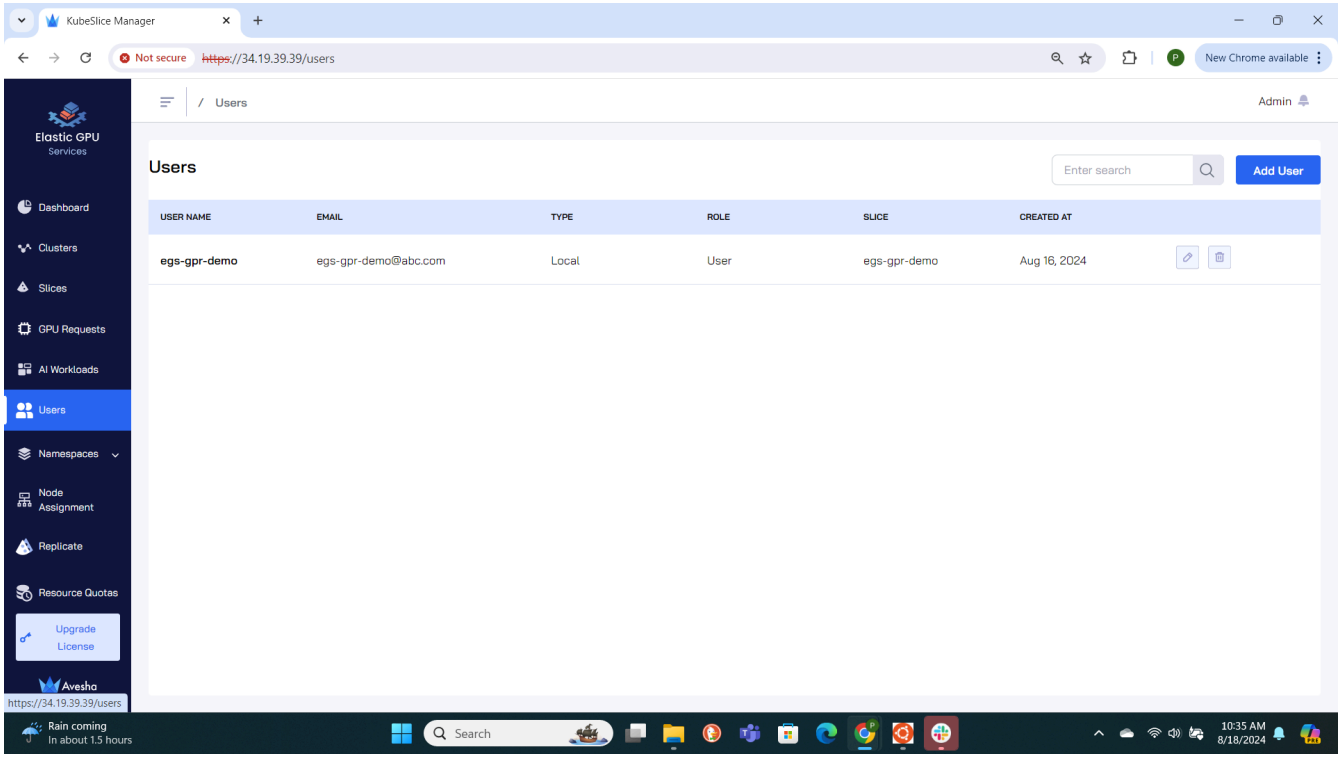


Add User to the Slice

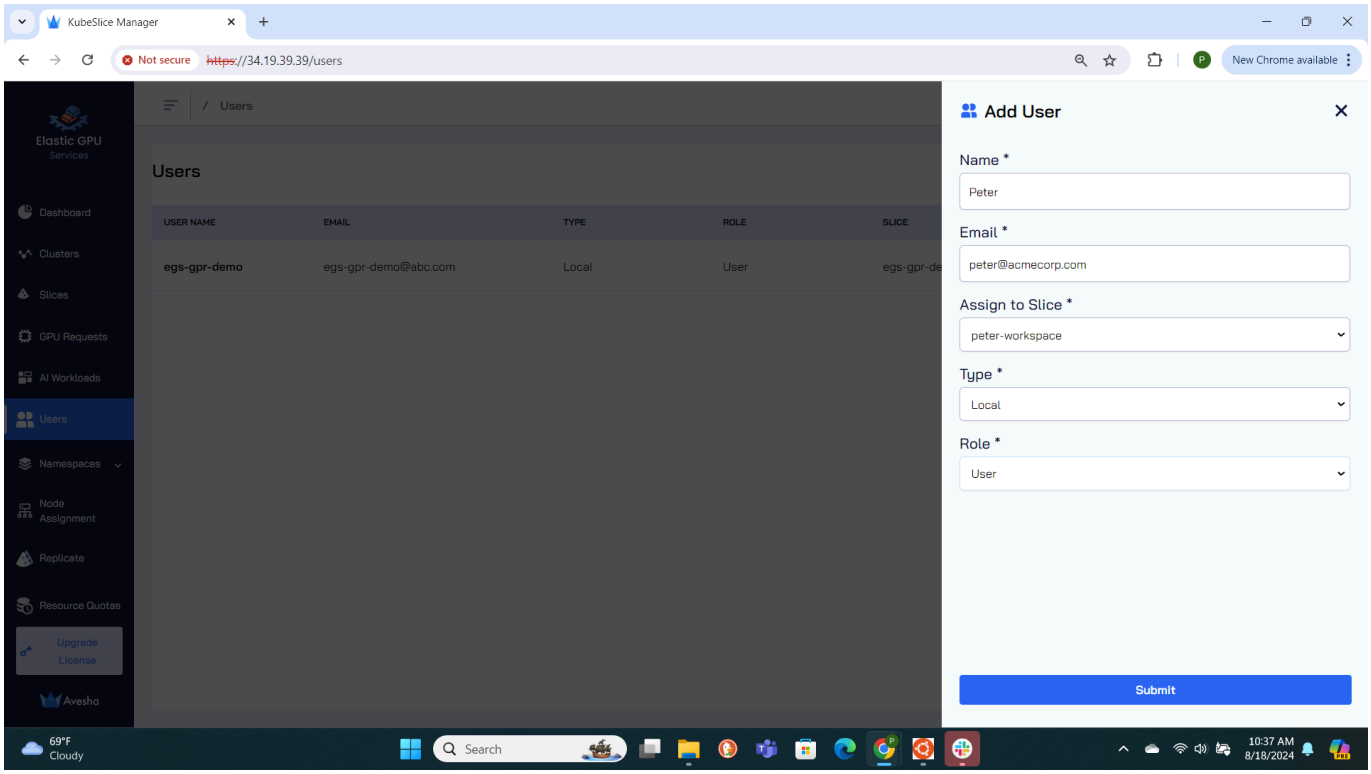
Admin can add, remove, or edit user details in the **Users** panel.

To add a User to the Slice, perform the following steps.

Select **Users** on the left panel.

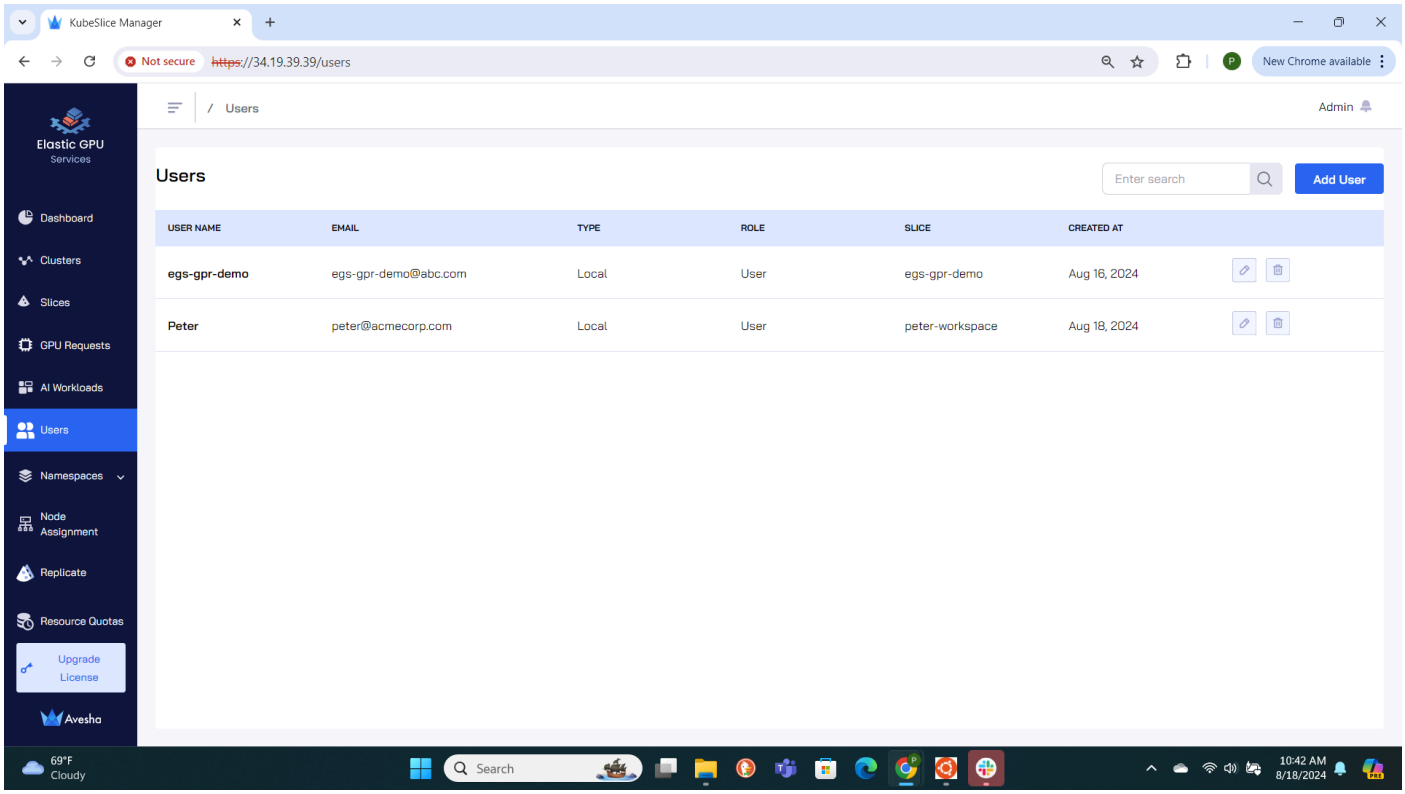


1. Click **Add User.**
2. Add the following user details:
    a. Name
    b. Email ID
    c. Select the slice name
    d. Select type `local` (or `ldp group` - will be supported soon)
    e. Select the **User** role.
    f. Click **Submit**.”



You should see the newly added user details in the **Users** list.

Elastic GPU Service

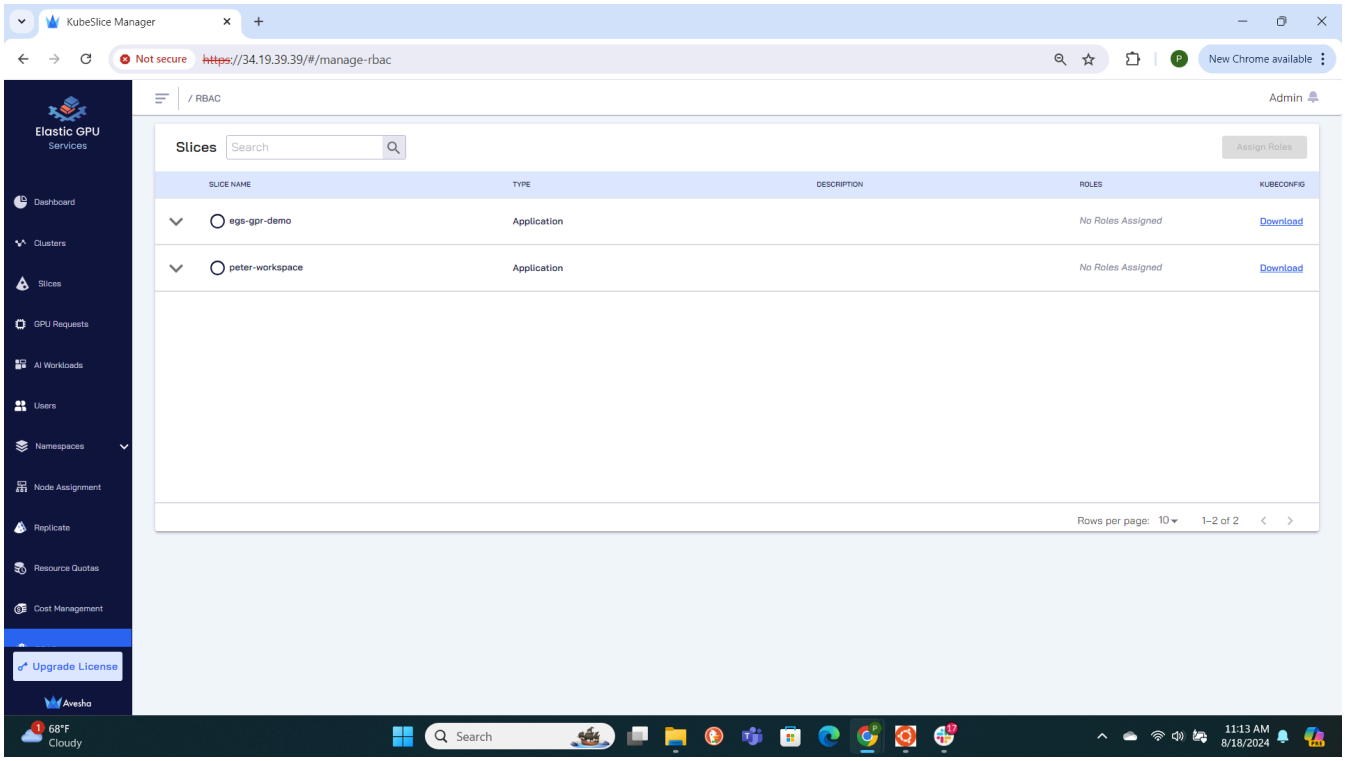


Provision Slice RBAC for User

Perform the following steps to create an RBAC role for user that restricts the user access to worker cluster user namespaces.

RBAC for Slice

1. Select **RBAC** on the left panel
2. Select the User Slice.
3. Click **Assign Roles**



Add Role Details

1. Select **Service Account** for the User/Group.
2. Enter a name for the Service Account.

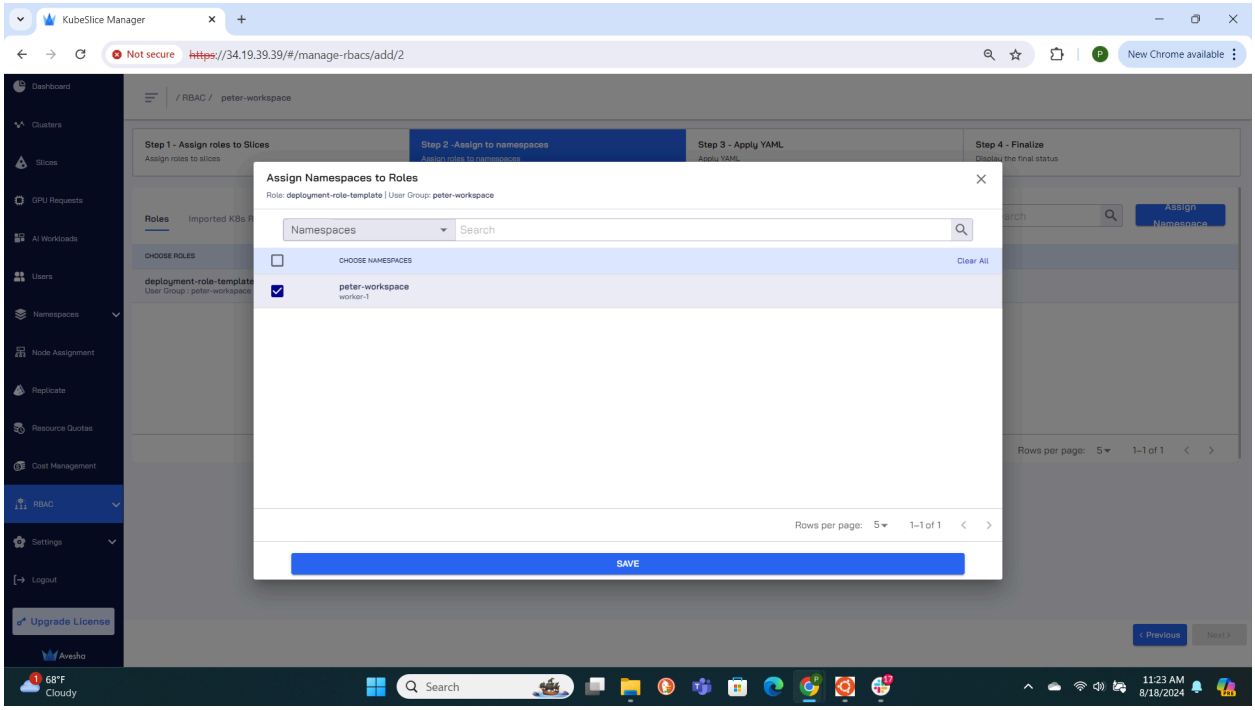
3. From the**Select roles** drop-down list, pick **deployment-role-template.**
4. Click **Next.**

   📒**Note**: Admin can edit/modify the default role templates, add new roles, and use them.

   For more details on the RBAC and roles, please see the RBAC documentation.

Go to the Step 2: Assign Namespace tab and:

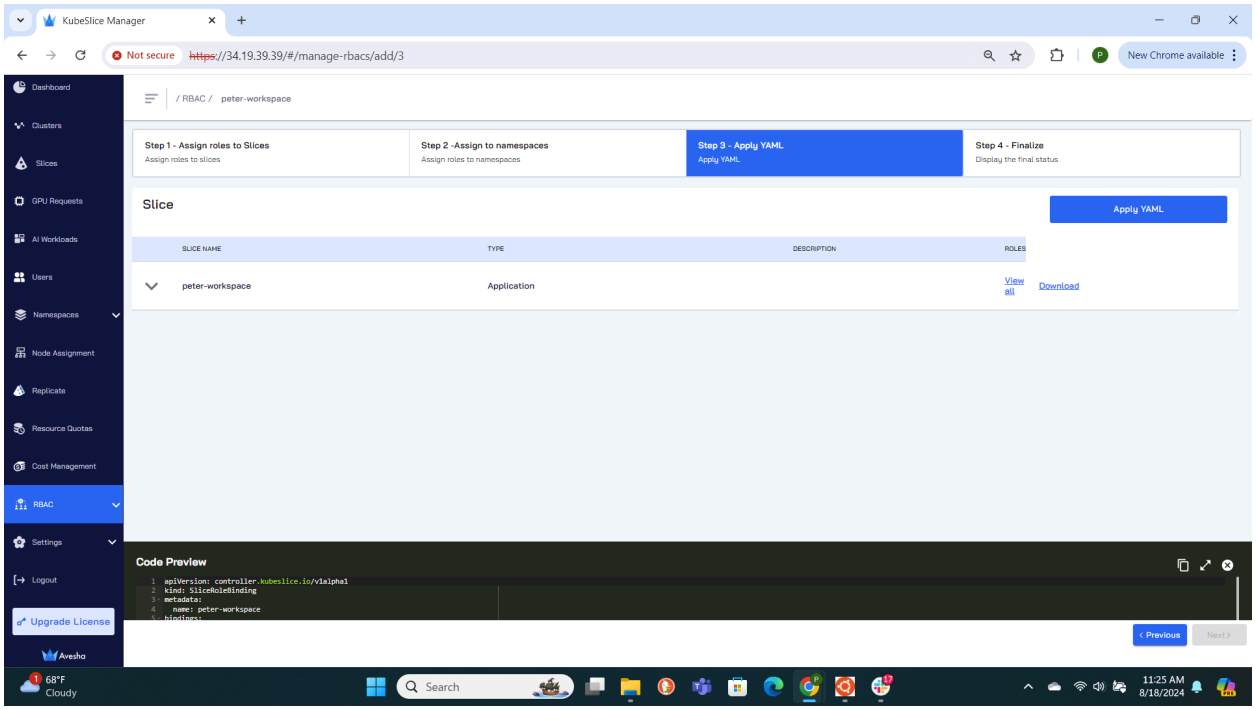
1. Select role from the list.
2. Click on Assign Namespace.
3. Select namespace.
4. Click **Save.**
5. Click **Next.**

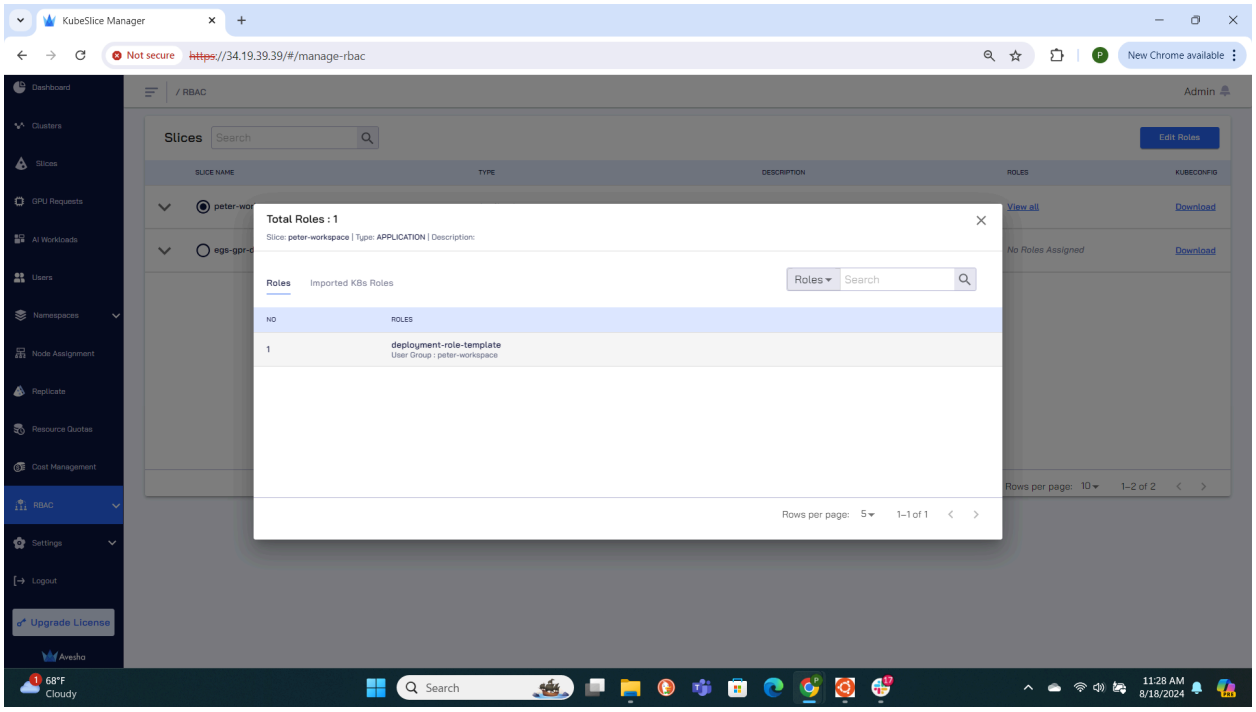


Go to the **Step 3 Apply YAM**L tab and:

1. Click **Apply YAML**
2. Click **Done** to apply the configuration.

Elastic GPU Service





User Access Token for UI Access

Users need an Access token to access the UI portal.
Perform the following steps on the cluster to generate the access token.

```
Unset
%./fetch_egs_slice_token.sh -h
%./fetch_egs_slice_token.sh -k <kubeconfig> -s <slice name> -p avesha

Note: default project name is "avesha" - change for different projects
```

Download KubeConfig for User

Admin can download the User RBAC Kubeconfig YAMLfile from the RBAC panel.

1. Select RBAC on the left panel.
2. Select the User Slice.
3. Click the **Download** button to download the user namespace-scoped Kubeconfig for the worker cluster.

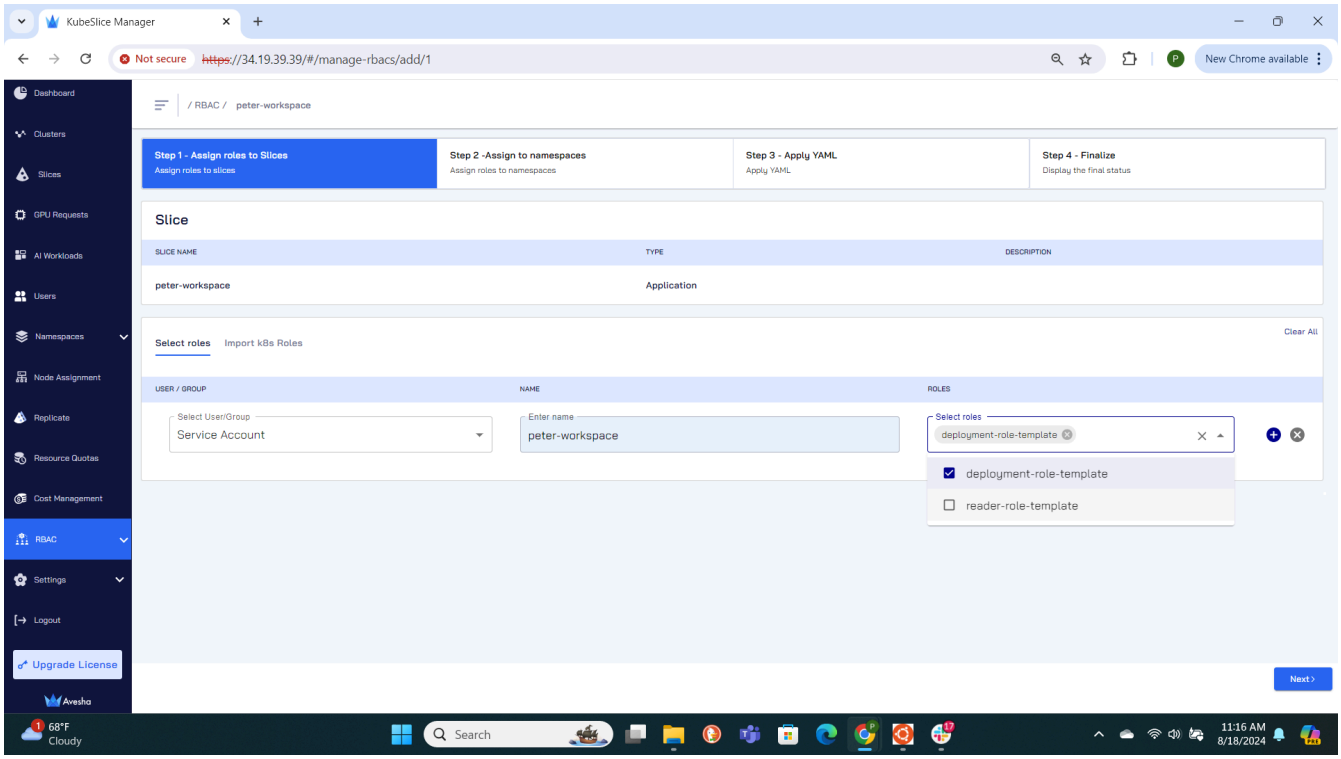
📒**Note**: User Slice RBAC role needs to be provisioned to download the KubeConfig YAML file.

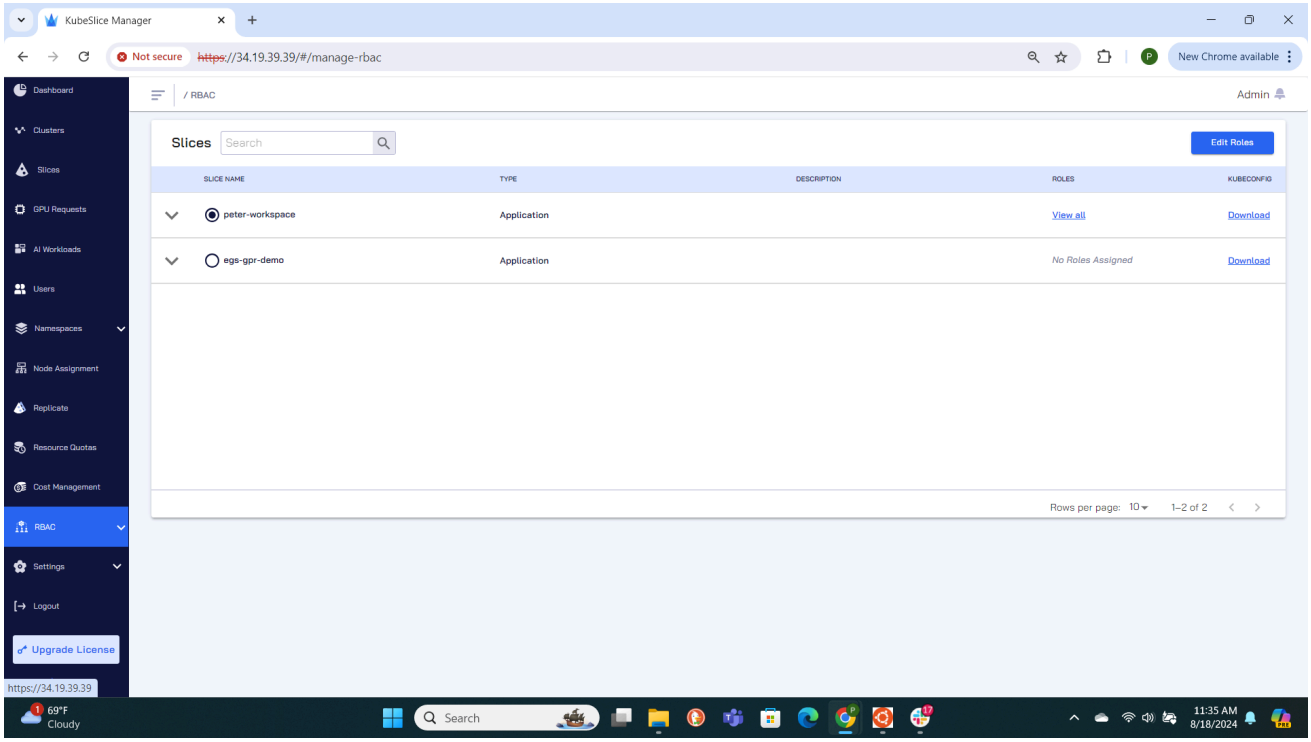
Go to the Slice page, on the **Slice** bar, click **view_all** to see the roles for the Slice.

Click the **Download** button to download the Kubeconfig file on the worker cluster with user-specific RBAC.

Share the Token with the User

Send the EGS UI portal (ui-proxy) URL and the token to the user to access the User Portal.





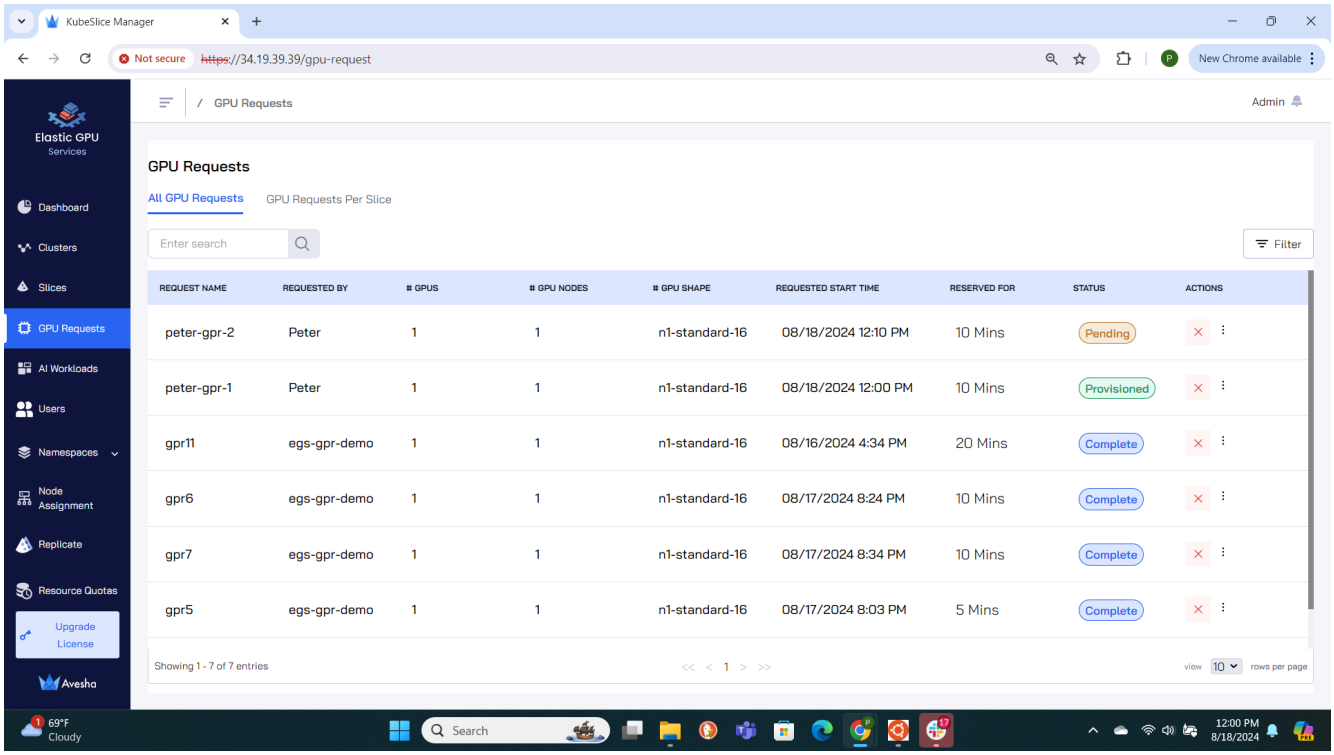
Manage GPR Queue

Admin can view and manage the User GPU provision requests (GPRs) queue.

View GPRs across all the Users

1. Select **GPU requests** on the left sidebar.The **All GPU Requests** tab shows all the GPU requests across all the Slices.

   Admin can use **search textbox or Filter** to filter the GPRs.



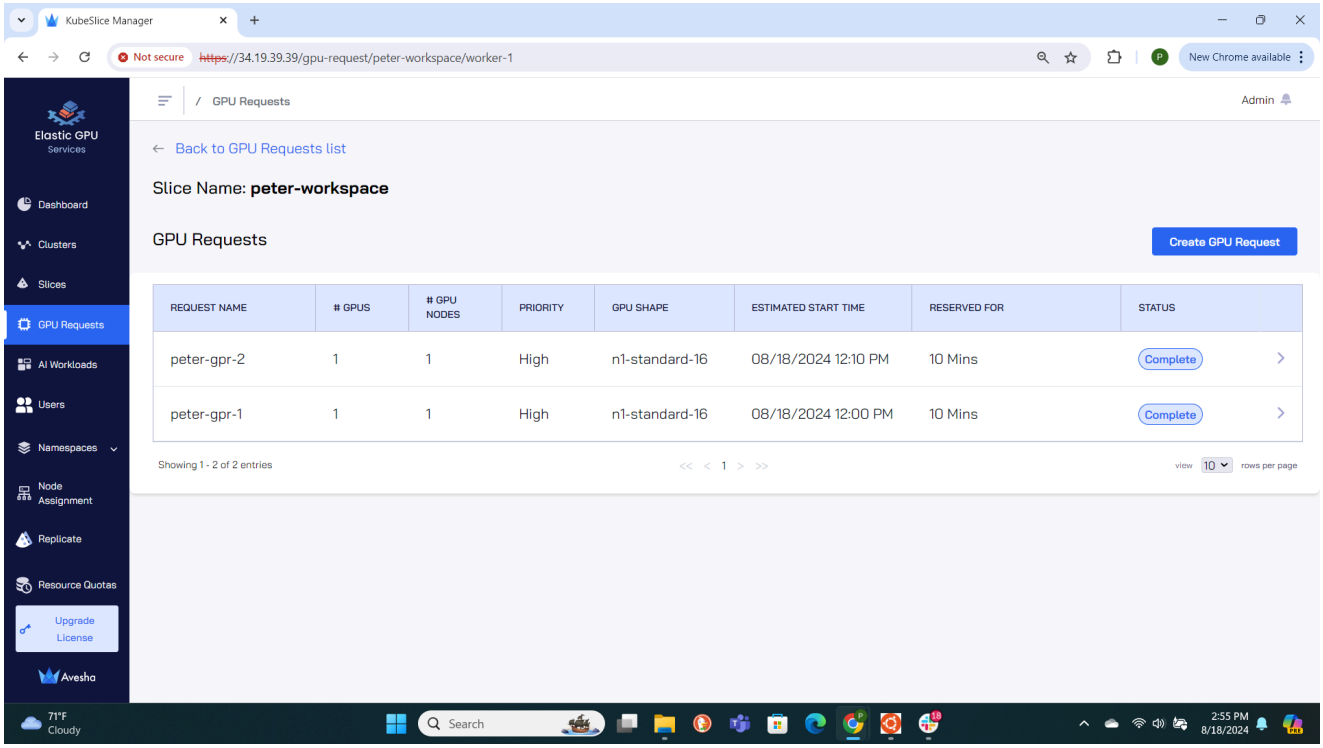
View the GPRs Specific to a Slice

Select the **GPU Requests Per Slice** tab, and click the Slice to see its specific GPRs.

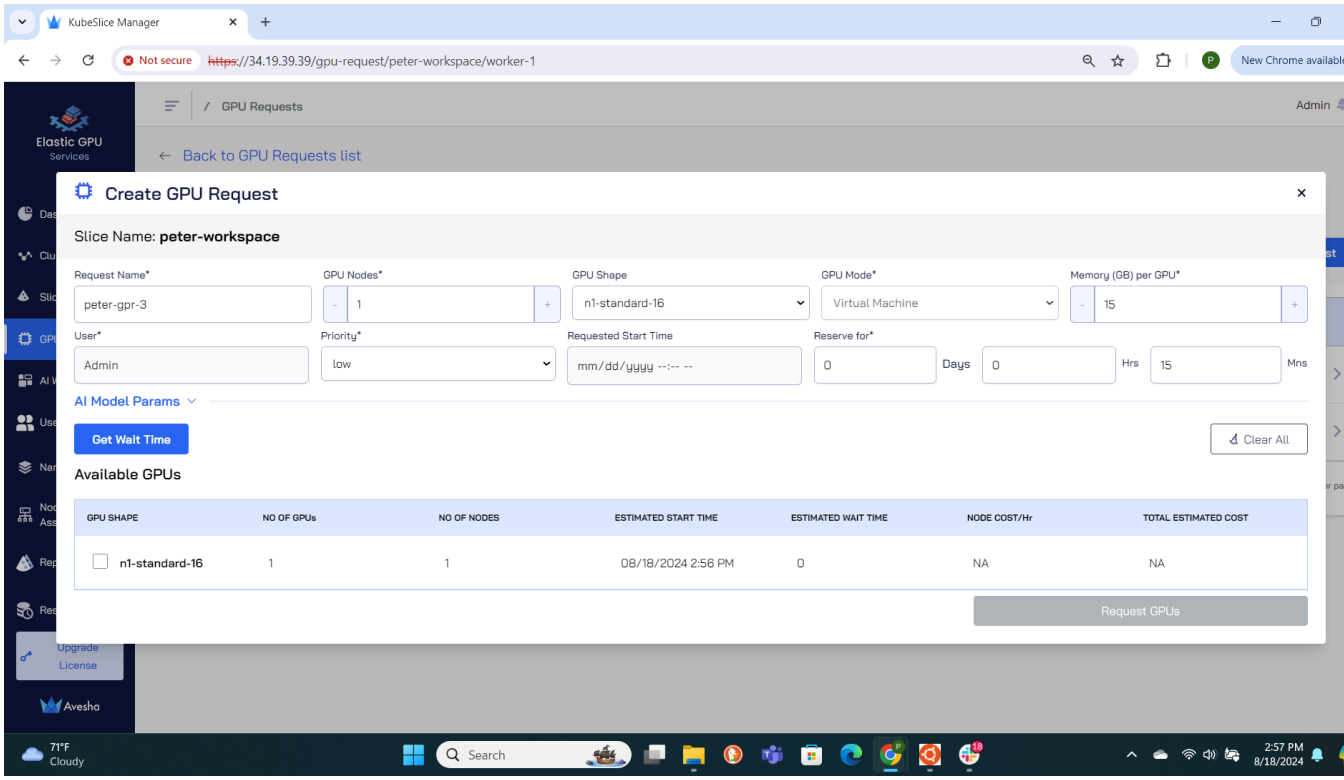
GPR Request for a User

Admins can create GPRs on behalf of the Users. Perform the following steps to create a GPR:

1. Select the User Slice in **GPU Requests Per Slice** tab.
2. Select the User Slice.
3. Click **Create GPU Request**.



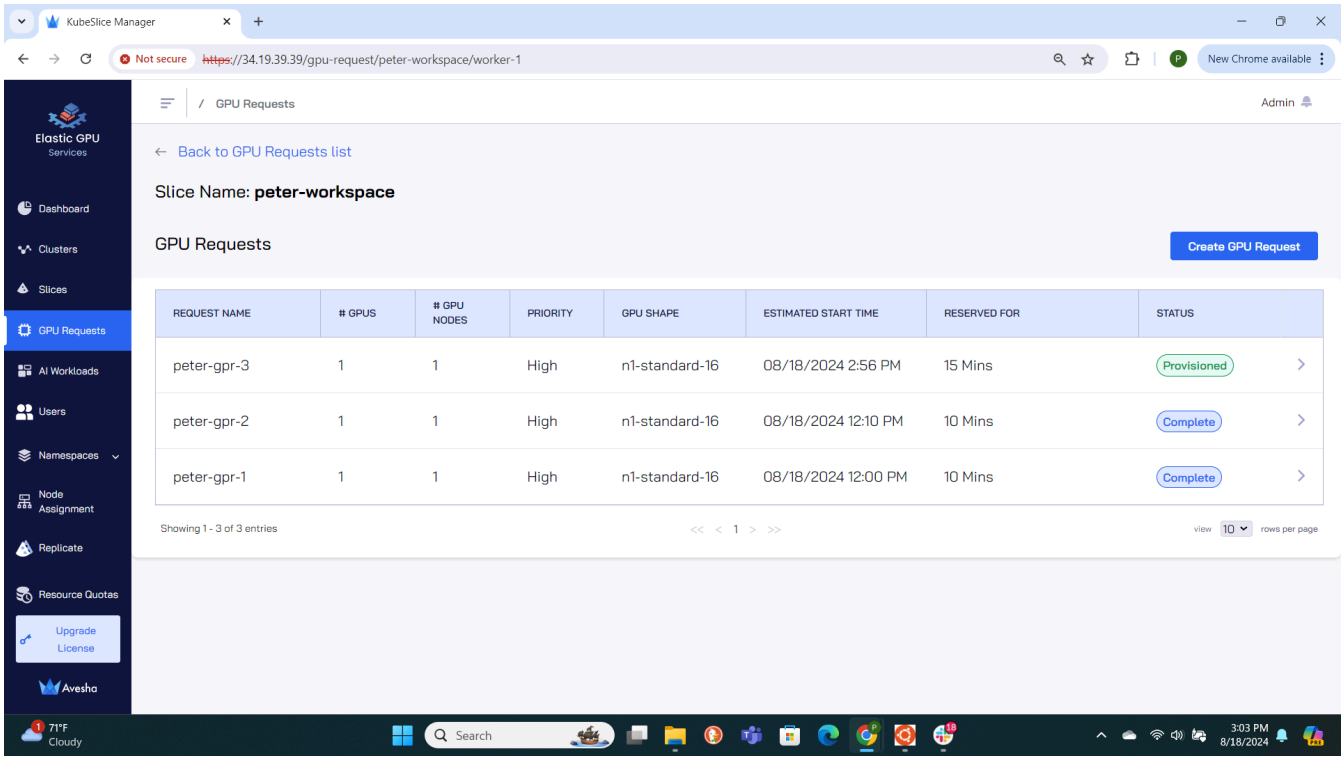
1. On the **Create GPU Request** page, add the request details:
2. Enter GPR name, Number of GPU nodes.
3. Select GPU shape.
4. Select priority.
5. Select Reserve for duration.
6. Click the **Get Wait Time** button. EGS shows the estimated wait time for the GPU nodes provisioning.
7. Select the GPU in the **Available GPUs** table with acceptable estimated wait time.
8. Click **Request GPUs**.



View the **GPR** in the **User's GPU Requests** queue.

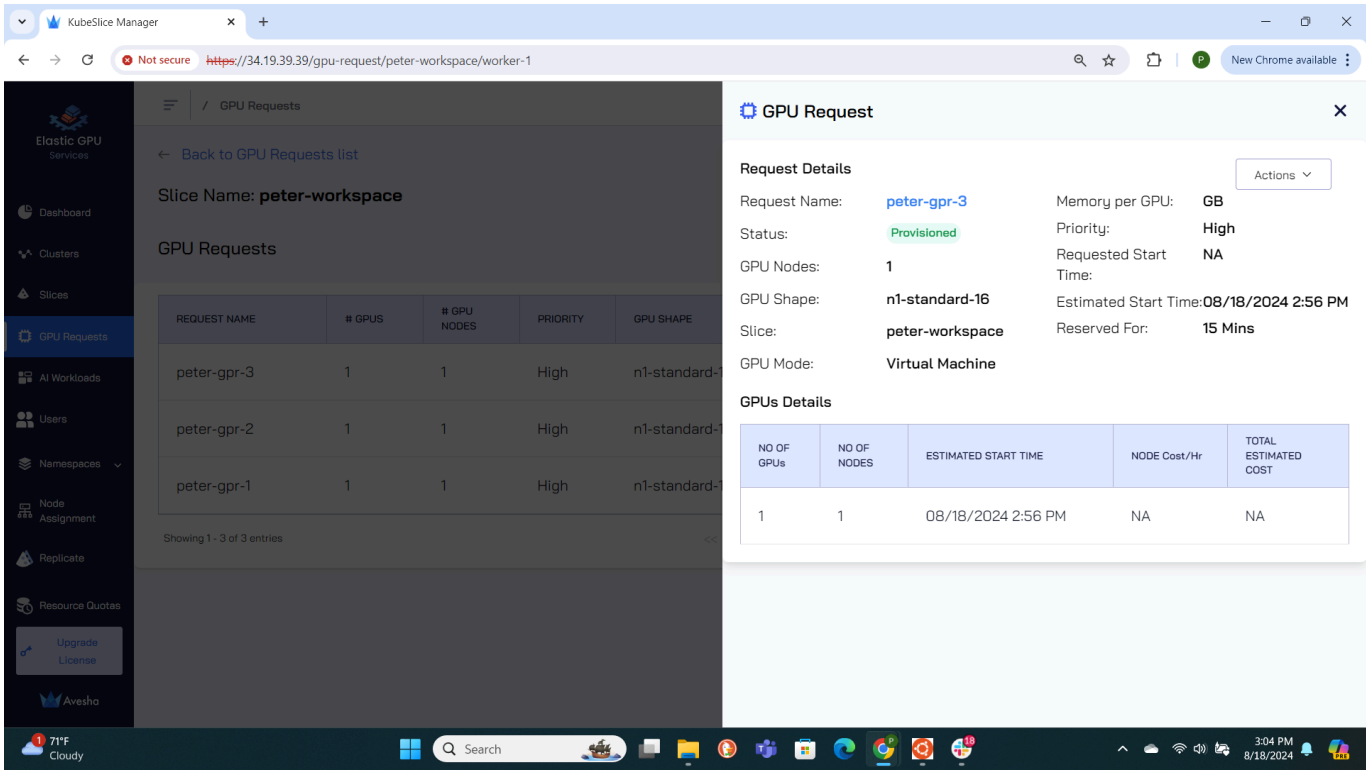
Note: GPRs can be created by additional methods using EGS APIs or by applying GPR custom resource YAML to the KubeApi server. These methods can be invoked by the CI/CD or RAG pipelines or external system/services or application services in the cluster.

Elastic GPU Service



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

Elastic GPU Service



Additional Operations

Admin can perform the following operations on the GPR queue.

GPR tables, Queues View and Actions

1. Admin can view the GPR queue.
2. Admin can change the priority of GPR and move a GPR up/down the queue.
3. Admin can edit, delete, or early-release a GPR
4. Admin can do GPR eviction.

Adjust GPR Priority

Admin can adjust the priority of a GPR in the queue. Admin can select a GPR and increase the priority number (low 0-100, medium: 100-200, high: 200 and above) to move a GPR higher in the queue. When a GPR is moved to the top of the queue, it will be provisioned when the resources are available to provision the GPR.

GPR Eviction

Admin can early release provisioned GPRs and make required nodes available for the high priority top GPR to be provisioned.

EGS shows Admin a list of GPRs that needs to be evicted to provision the top GPR. Admin can manually early-release the GPRs to make room for the top GPR.

Approve GPRs in Pending-Approval State

Admin will approve the GPRs that are in pending approval state. A GPR will be in pending approval state for any of the following conditions:

1. Exceed the quota for the Slice
2. Exceed the GPU nodes limit per GPR (request for large number of GPU nodes)
   📒**Note**: Most GPRs are auto-approved

Admin will use the Actions pull down menu to *approve/deny* the GPR.

Early Release a Provisioned GPR

Admin can early-release a provisioned GPR. Early release of a GPR will remove the associated GPU nodes from the Slice VPC.

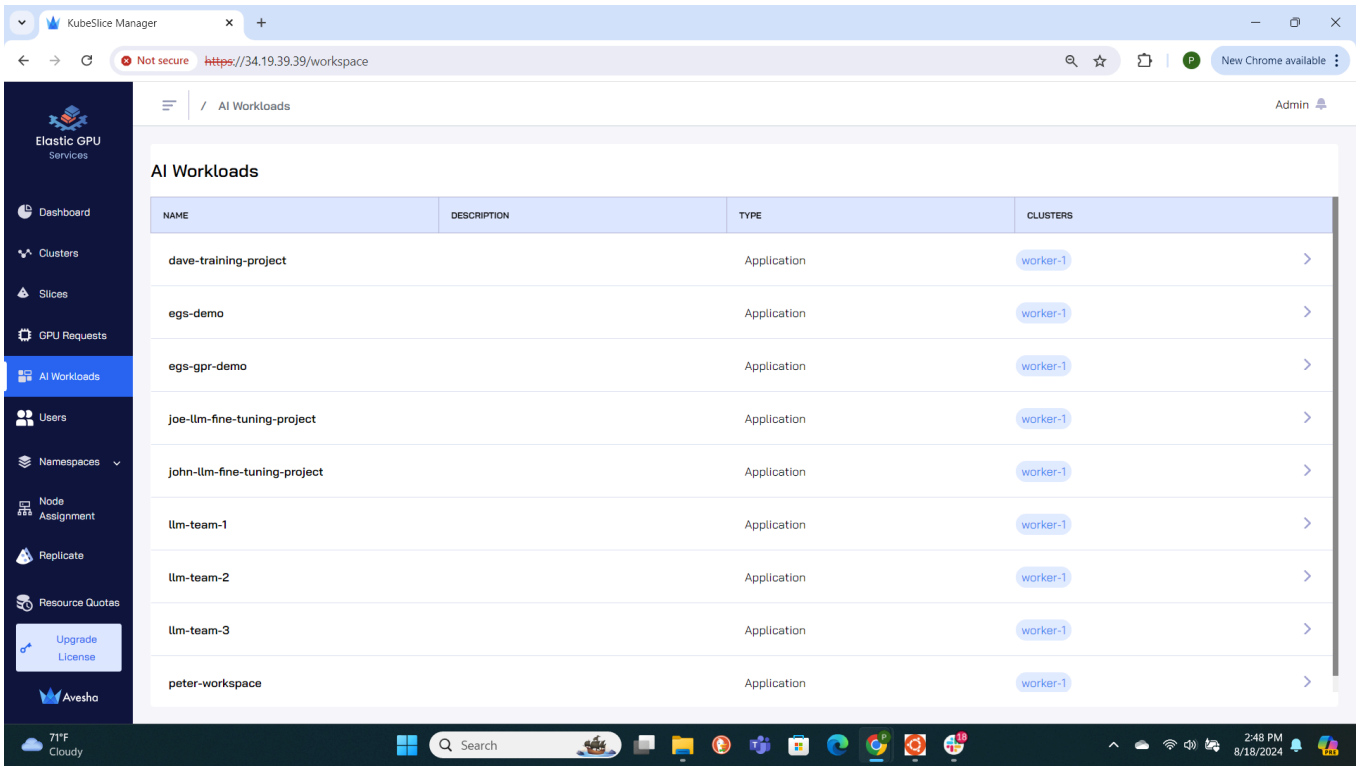
Admin can use this workflow to free up GPUs to provision a higher priority GPR. Admin can use this workflow for any other admin reasons or under utilization of GPU resources, User/Manager request and so on.

📒**Note**: Adjust priority and GPR eviction workflows will be available in the next release.

View AI Workloads

Admin can view all Slice workspaces AI workloads.

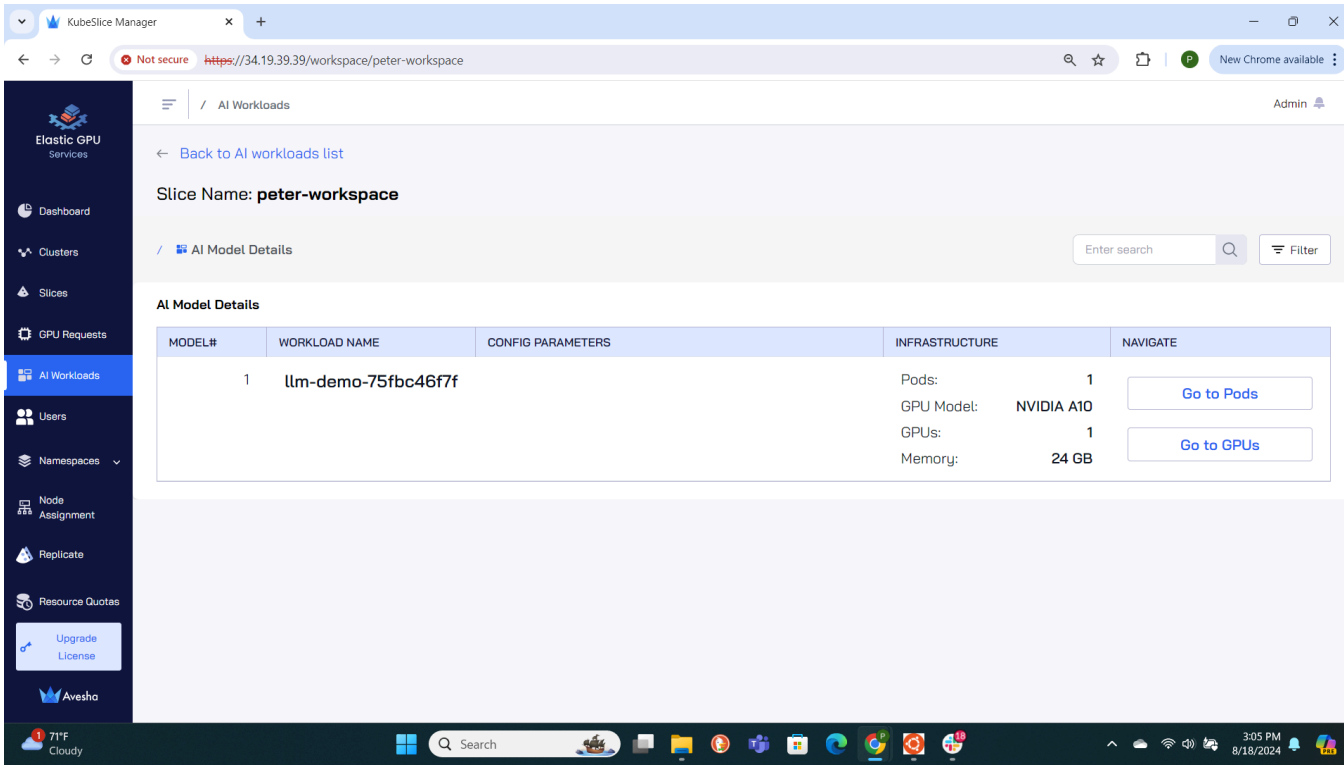
1. Select **AI Workloads** from the left panel.
2. Select User Slice to see the AI workloads for the slice.



Model Details

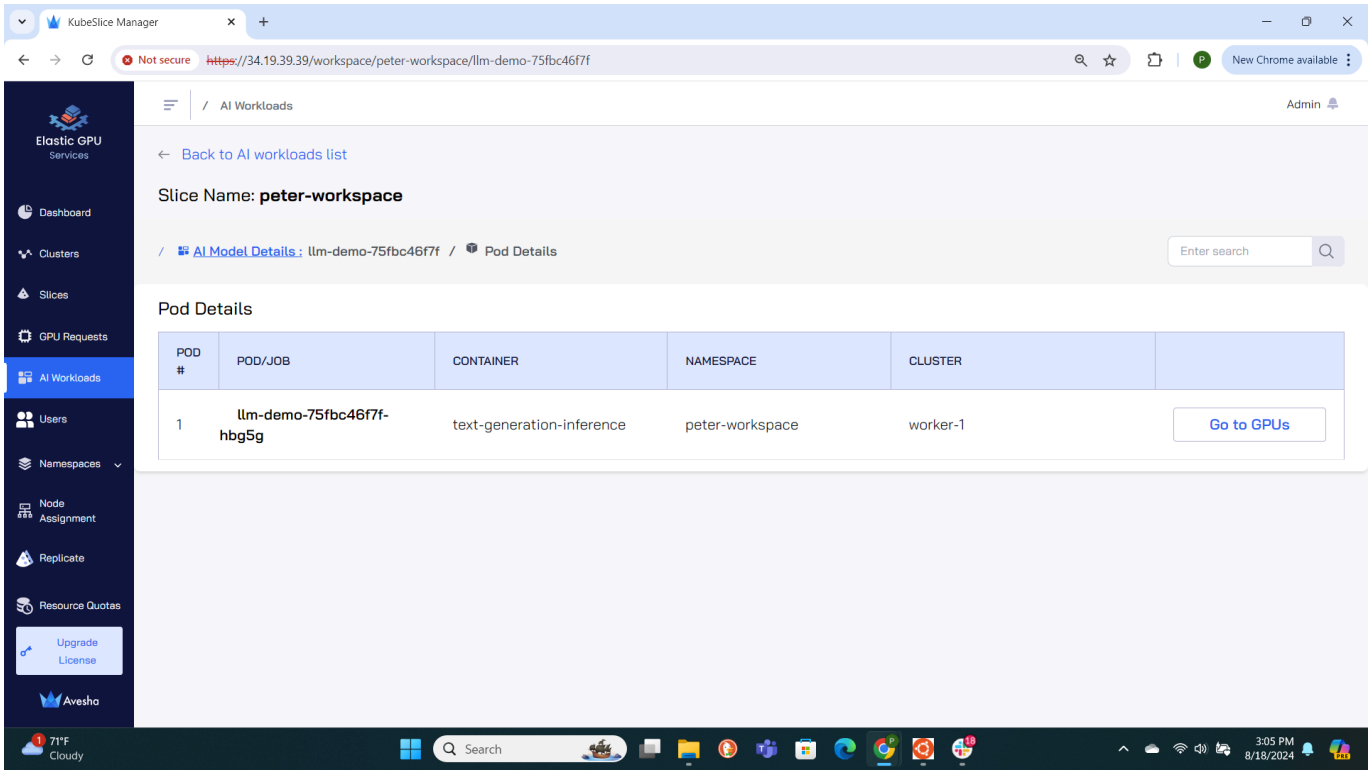
Select User Slice to view AI workloads for User workspace. The model details:

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



View Pods

Click **Go to Pods** to view the pods running with GPUs in the workspace.



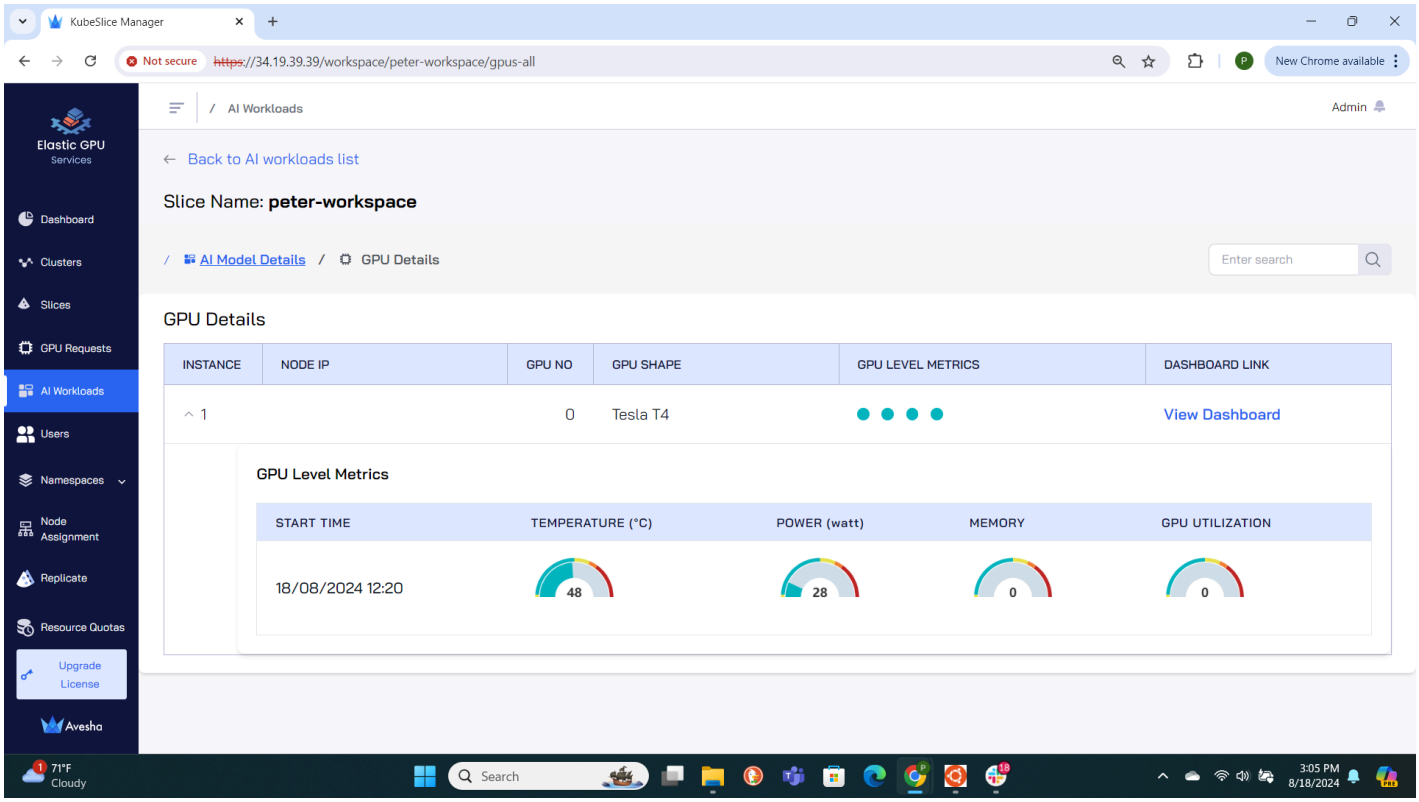
View GPUs

On the **AI Model Details** page, click **Go to GPUs** to view 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 time-series data for the selected GPU device.

Elastic GPU Service



Alerts and Events

Create Project

During installation a default project workspace will be created. A single project namespace can be used to manage one or more clusters across one or more users/teams. In a project one or more slices can be created. A user can be associated with one or more slices. Each slice provides a workspace for a user or a team. Additional projects can be created to provide multi-tenancy across organizations or large departments, where a pool of clusters are managed by the different departments.

For more details on how to create projects - please see <link>

Registers Clusters

During an installation performed using the installation script for single cluster deployment, the script registers the worker cluster.

Admin can add additional clusters to the EGS control plane to manage the GPU resources on the clusters.

For more details on how to register a cluster please see <link>

Admin Dashboard

1. Use the dashboard to monitor key metrics related to the GPU pool.
    - GPRs and GPU utilization, allocation, Errors, Alerts
2. Use the dashboard to access Cluster kubernetes dashboard pages.

Day-to-Day Operations

Admin can perform the following operation for managing the day-to-day operations to manage the EGS platform.

1. Admin can add/edit/delete User Slices (workspaces).
2. Admin can add/remove namespaces to Slice.
3. Admin can add/remove/update the Slice RBAC and regenerate the KubeConfig for User - for cluster access.
4. Admin can generate access tokens for Users for UI access
5. Admin can view GPRs queue across all the Users/Slices.
6. Admin can edit/delete GPRs - delete or early-release a provisioned GPR.
7. Admin can view AI workloads across all the Users/Slices.
    - View GPU dashboards for AI workloads. Access time-series data for all workload GPUs.
    - View pods, model details.
    - Access top power/temp GPUs. Filter/search GPUs.
8. Admin can register additional Clusters
    - add/remove clusters
9. Admin can create additional Projects
    - Add/remove projects

The following workflows will be available in the next release:

1. View admin Dashboards
    - Error/Warning alerts, top GPUs, top Users, top utilization, insights, etc.
2. Adjust GPR priority, GPR eviction - to make room for high priority GPR
3. Inventory schedule table views
4. Approve pending GPRs
5. View GPR pre-provisioning checklist/verification (visibility into the GPR pre-checks and logs)
    - pre-provision the nodes and run checks
        i. Node health check and Node PV/PVC check
        ii. Nodes network check - RDMA subnet check and NCCL latency check
        iii. Nodes connectivity check
6. View GPR post-exit operations (visibility into GPR post-checks and logs)
    - Node draining, Network check, NCCL test checks
    - PVC/PV check, Labels check
7. Enable provisioning of dynamic node pools and nodes for Slice VPCs.