



# 依瞳人工智能开发平台 用户手册

依瞳科技（深圳）有限公司

2020.6.20

[www.apulis.com](http://www.apulis.com)

广东省深圳市南山区粤海街道高新南九道微软科通大厦 18D

## 目录

1.	用户指南 .....	1
2.	产品简介 .....	1
2.1.	产品概述 .....	1
2.2.	名词解释 .....	1
2.3.	功能简介 .....	3
3.	操作说明 .....	3
3.1.	注册和登录 .....	3
3.1.1	账号密码注册登录 .....	4
3.1.2	微信注册登录 .....	4
3.1.3	微软邮箱注册登录 .....	5
3.2.	平台主页 .....	5
3.2.1	菜单栏 .....	6
3.2.2	主页图表 .....	7
3.3.	Submit Training Job .....	9
3.3.1	基本设置 .....	9
3.3.2	高级配置 .....	12
3.3.3	模板设置 .....	14
3.4.	View and Manage Jobs .....	14
3.4.1	job 列表 .....	15
3.4.2	job 详情 .....	16
3.5.	Cluster Status .....	18
3.5.1	Team Virtual Cluster Status .....	19
3.5.2	Team VC User Status .....	19
3.5.3	Cluster Usage .....	21
3.5.4	Physical Cluster Node Status .....	21
3.6.	Virtual Cluster .....	22
3.6.1	VC 列表 .....	23
3.6.2	新增 VC .....	23
3.6.3	修改 VC .....	24
3.6.4	删除 VC .....	25
3.7.	USER DASHBOARD .....	25
3.7.1	顶端菜单栏 .....	26
3.7.2	Dashboard .....	27
3.7.3	User .....	27
3.7.4	Group .....	30
3.7.5	Role .....	32
3.8.	报警配置 .....	34
3.8.1	config.yaml .....	35
3.8.2	ecc-config.yaml .....	36
3.8.3	rule-alerts.yaml .....	37
3.8.4	rule-config.yaml .....	37

## 1. 用户指南

在本文档中，您可以了解到用户在依瞳人工智能开放平台上，进行深度神经网络模型训练的操作方法，并且可以管理自己的训练任务、监控任务的执行情况、查看集群节点的运行情况等。

目标读者：深度学习工程师和使用者。

## 2. 产品简介

### 2.1. 产品概述

依瞳人工智能平台旨在为不同行业的用户提供基于深度学习的端到端解决方案，使用户可以用最快的速度、最少的时间开始高性能的深度学习工作，从而大幅节省研究成本、提高研发效率，同时可为中小企业解决私有云难建成、成本高等问题。

平台融合了 Tensorflow、PyTorch、MindSpore 等开源深度学习框架，提供了模型训练、超参调优、集群状态监控等开发环境，方便 AI 开发者快速搭建人工智能开发环境，开展 AI 开发应用。在监控模块基础上搭建预警模块，自动将平台异常通知管理员，提升平台的预警效率及安全性能。

平台底层采用更轻量级的虚拟化技术，如 Docker 容器，将任何一个或多个程序封装起来，并为容器提供标准的管理接口，使得每个容器之前互相隔离、互不影响，从而区分计算资源。对部署容器化的应用，采用 Kubernetes 集群技术，进行自动化部署、规划、更新和维护，避免运维人员进行复杂的手工配置和处理，从而提高效率，降低成本。

### 2.2. 名词解释

术语、缩略语	解释
Tensorflow	TensorFlow 由 Google 大脑主导开发，是一个分布式系统上的大规模深度学习框架。移植性好，可以运行在移动设备上，并支持分布式多机多卡训练，支持多种深度学习模型。
PyTorch	由 FaceBook AI 团队主导开发。不同于 TensorFlow，PyTorch 采用动态计算图的方式，并提供良好的 Python

	接口，代码简单灵活，使用起来非常方便。内存分配也经过了优化，能支持分布式多机训练。
MindSpore	MindSpore 是端边云全场景按需协同的华为自研 AI 计算框架，提供全场景统一 API，为全场景 AI 的模型开发、模型运行、模型部署提供端到端能力。
Docker	Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的镜像中，然后发布到任何流行的 Linux 或 Windows 机器上，也可以实现虚拟化。
Kubernetes	简称 K8S，是一个开源的，用于管理云平台中多个主机上的容器化的应用，Kubernetes 的目标是让部署容器化的应用简单并且高效，Kubernetes 提供了应用部署，规划，更新，维护的一种机制。
超参数	在机器学习的中，超参数是在开始学习过程之前设置值的参数，而不是通过训练得到的参数数据。通常情况下，需要对超参数进行优化，给学习机选择一组最优超参数，以提高学习的性能和效果。
Job	模型训练任务
VC	Virtual Cluster 虚拟集群，对物理集群内所有 AI 计算芯片进行分组管理，每一个组就是一个虚拟集群。
物理节点	表示集群中的物理机器
AI 计算芯片	用于 AI 模型训练所需要的处理器，比如英伟达 GPU
镜像	执行模型训练所需要的文件集合
Jupyter	Jupyter Notebook 是一个基于 Web 的交互式计算环境，支持运行多种编程语言。平台支持使用 Jupyter Notebook 的方式进行算法代码编写，模型训练任务的提交，以及结果查看等操作。
SSH	是一种加密的网络传输协议，可以远程登录到 job 容器。

TensorBoard	TensorBoard 是一个可视化工具，它可以用来展示网络图、张量的指标变化、张量的分布情况等。在训练网络的时候，我们可以设置不同的参数（比如：权重 W、偏置 B、卷积层数、全连接层数等），使用 TensorBoard 可以很直观地进行参数的选择。
-------------	---

## 2.3. 功能简介

本文主要介绍平台的功能和使用，包括提交 job、查看和管理 job、集群状态监控、虚拟集群管理、用户管理等。用户通过 web 端提交深度神经网络模型训练任务，通过查看和管理 job 页面可查看任务的运行状态、实时的资源使用情况、训练任务的日志输出等；通过集群状态监控可查看整个集群的资源使用情况，并可监控各物理节点的状态。

## 3. 操作说明

### 3.1. 注册和登录

用户在浏览器地址栏输入平台地址，进入登录页面，默认为账号密码登录的页面。如图 1 所示。【注意】浏览器推荐 Chrome。

平台目前支持三种登录方式，账号密码登录、微信登录和微软邮箱登录。

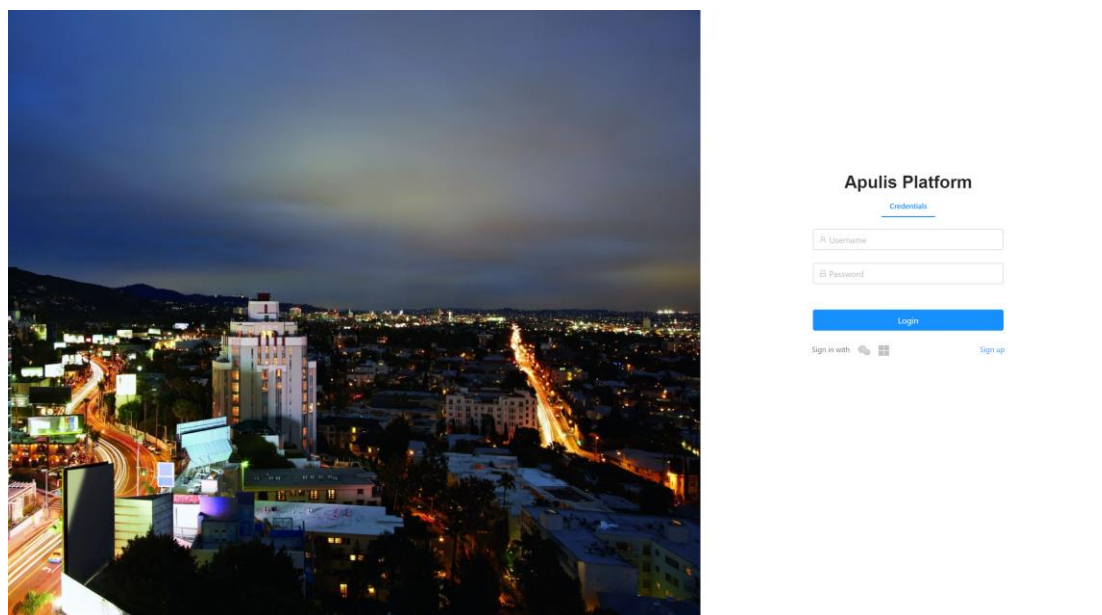


图 1 登录页面



### 3.1.1 账号密码注册登录

点击登录页面的 Sign up, 跳转到注册页面, 如图 2 所示。设置用户名、昵称、密码后即可完成注册, 完成注册后跳转到登录页面。使用新注册的账号登录平台后会提示“Sorry,you are no authorized to access this page”, 请联系系统管理员获取权限, 权限获取后, 则可以正常使用平台。

如用户已有账号, 在登录页面输入账号和密码后点击 login 按钮, 即可登录平台。

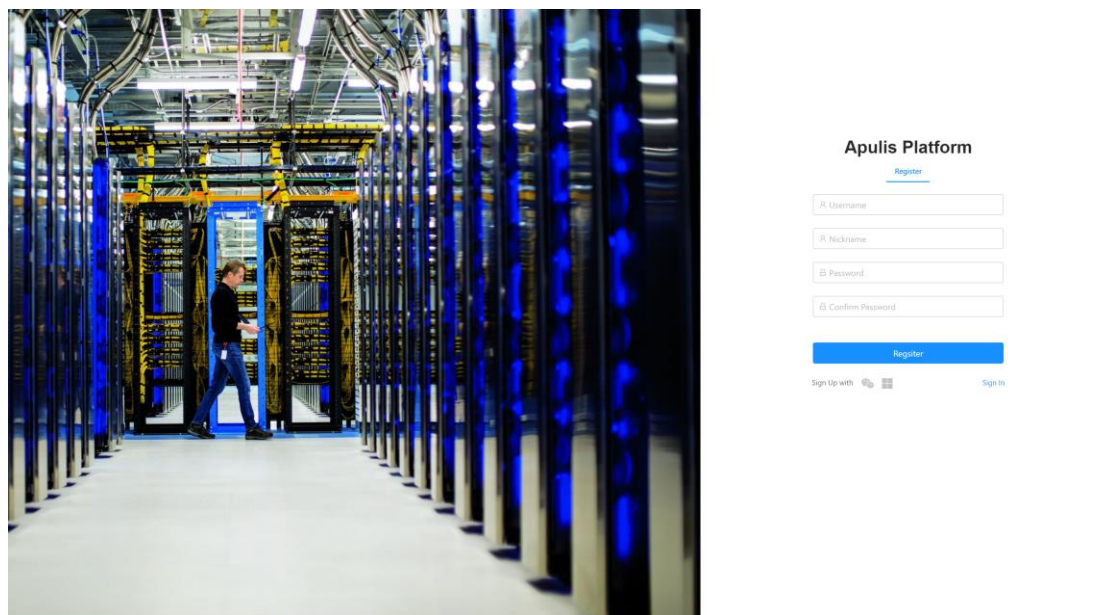


图 2 用户注册页面

### 3.1.2 微信注册登录

在登录页面选择“Sign in with 微信”后, 跳转到微信扫码页面。如果微信未注册, 扫码后跳转到微信注册页面, 如图 3 所示, 设置用户名、昵称、密码后即可完成注册。新注册的用户需联系管理员获取权限, 权限获取后, 则可以正常使用平台。如扫码的微信已注册, 扫码后即可登录平台。

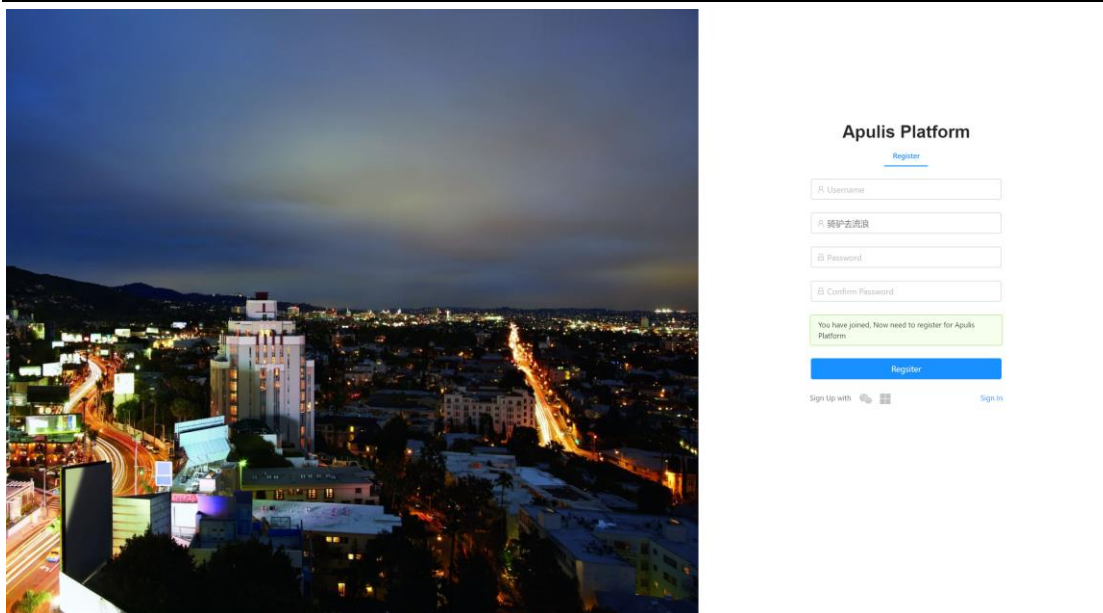


图 1 微信注册

### 3.1.3 微软邮箱注册登录

在登录页面选择“Sign in with 微软”后，跳转到微软邮箱登录页面，输入邮箱账号、密码后即可登录。第一次通过邮箱登录的账户，需联系系统管理员获取权限。

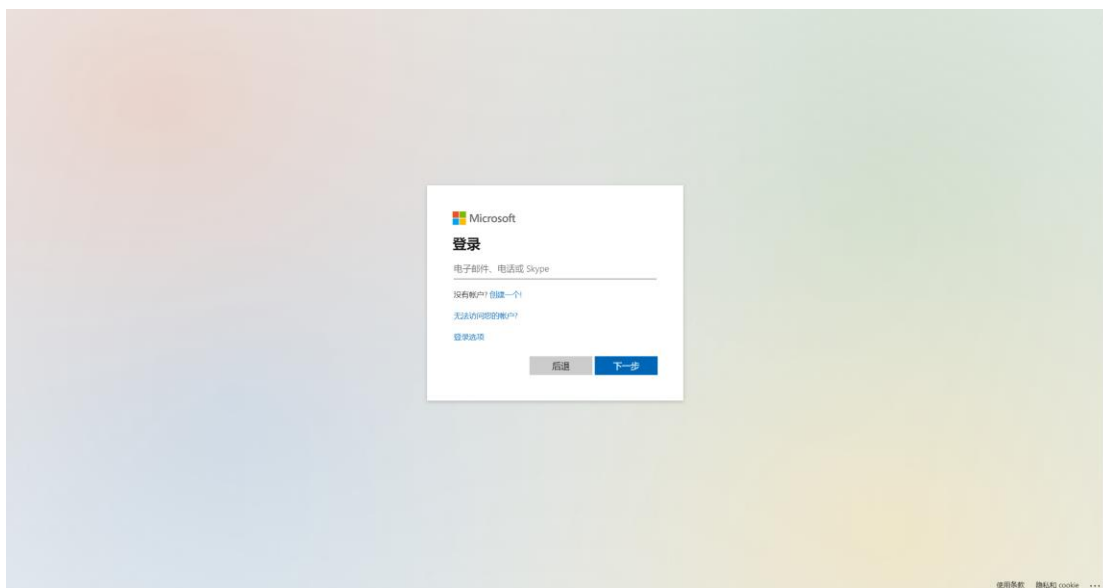


图 2 微软邮箱登录

## 3.2. 平台主页

页面顶端菜单栏，展示当前用户所在 VC 资源组、用户昵称、USER DASHBOARD 和退出。

左侧菜单栏包括 6 个菜单，分别为 Home（主页）、Submit Training Job（提交训练 job 页面）、View and Manage Jobs(查看和管理 job 页面)、Cluster Status(查看物理集群相关信息页面)、Virtual Cluster(查看虚拟集群相关信息页面)、Edge Inference（中心侧推理）。

主页图表中有资源统计和快捷跳转链接，见图 5。

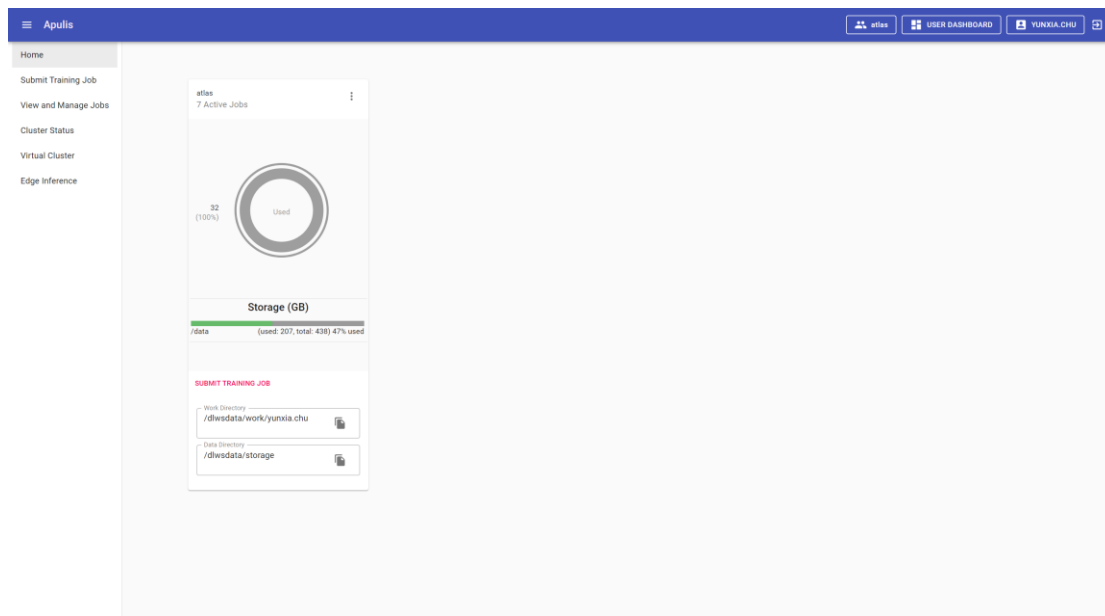


图 3 HOME 主页

### 3.2.1 菜单栏

页面顶端的菜单栏如图 6 所示，最左侧图标的功能为显示、隐藏左侧菜单栏；点击“Apulis”可以跳转到平台主页界面；“platform”为当前所选择的虚拟集群 VC 的名称，点击可进行切换其他 VC；点击菜单“USER DASHBOARD”可跳转到用户管理页面，为退出按钮，点击后则退出平台。



图 4 平台主页上方菜单栏



### 3.2.2 主页图表

主页的图表包括四部分内容，分别为当前 VC 中正在运行的 job 数量统计、资源使用情况统计、集群中的存储资源统计、展示工作路径和数据路径，便于用户了解 VC 资源的使用情况和物理集群的存储情况，见图 7。

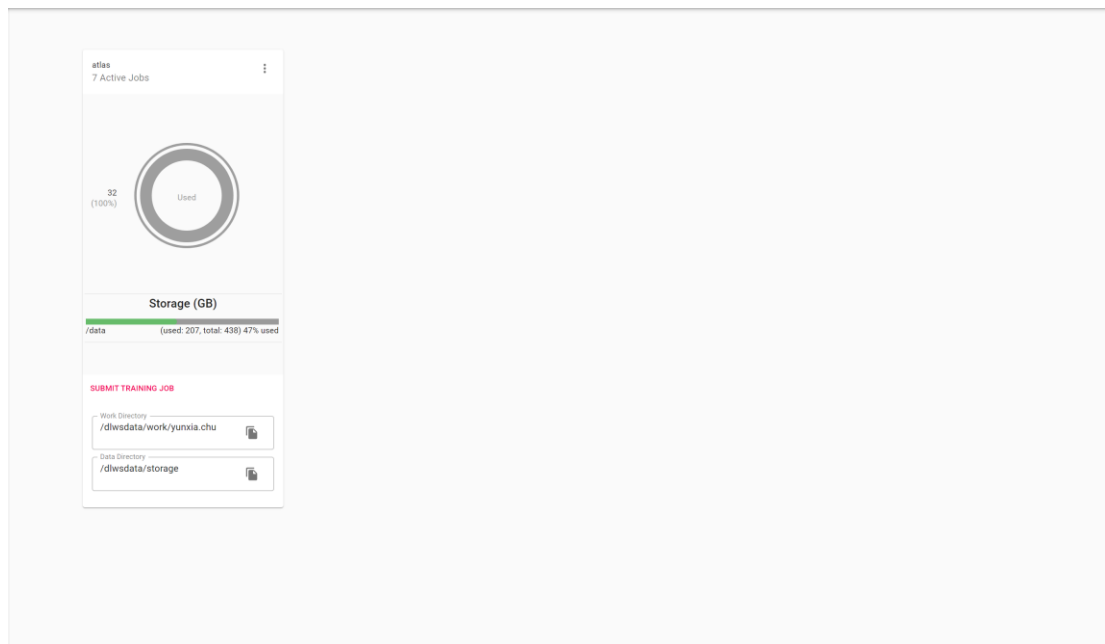


图 7 首页图表统计

除统计和路径展示外，也提供了提交 job、查看 job 和查看集群状态的跳转链接。点击图 7 中的“SUBMIT TRAINING JOB”，即跳转到 submit training job 页面。点击图表右侧的

⋮ 后，如图 8 所示，选择“Cluster Status”，即跳转到 Cluster Status 菜单的 TEAM VIRTUAL CLUSTER STATUS 页签，如图 9 所示，可查看虚拟集群中的资源和 job 运行情况；选择“View Jobs”，即跳转到“View and Manage Jobs”菜单的 MY JOBS 页签，如图 10 所示，可查看当前登录账户下的所有 job 列表。

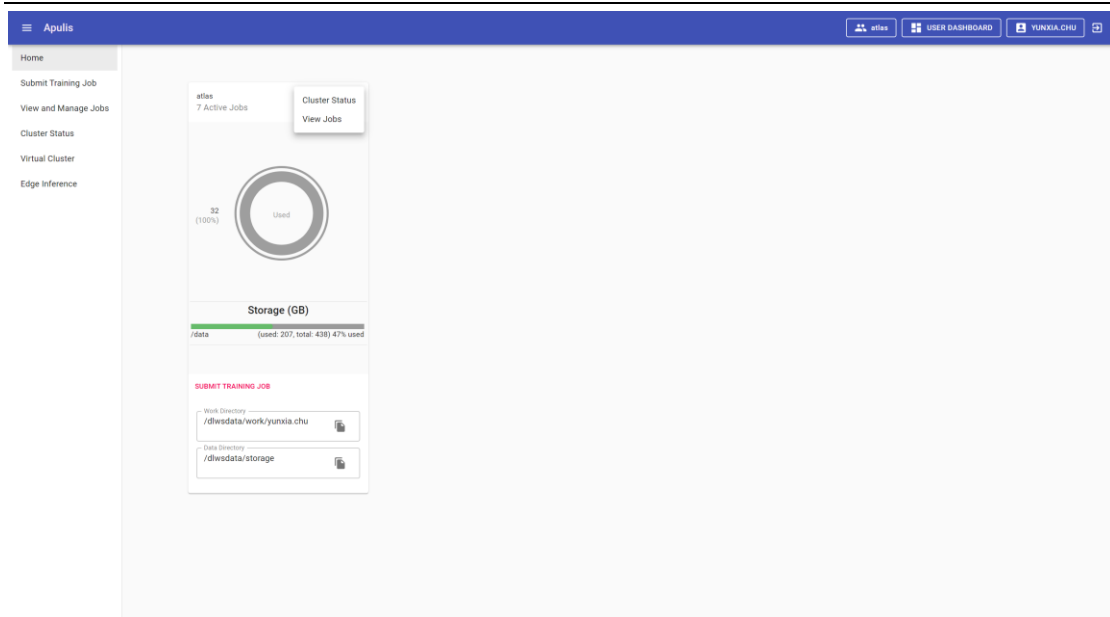


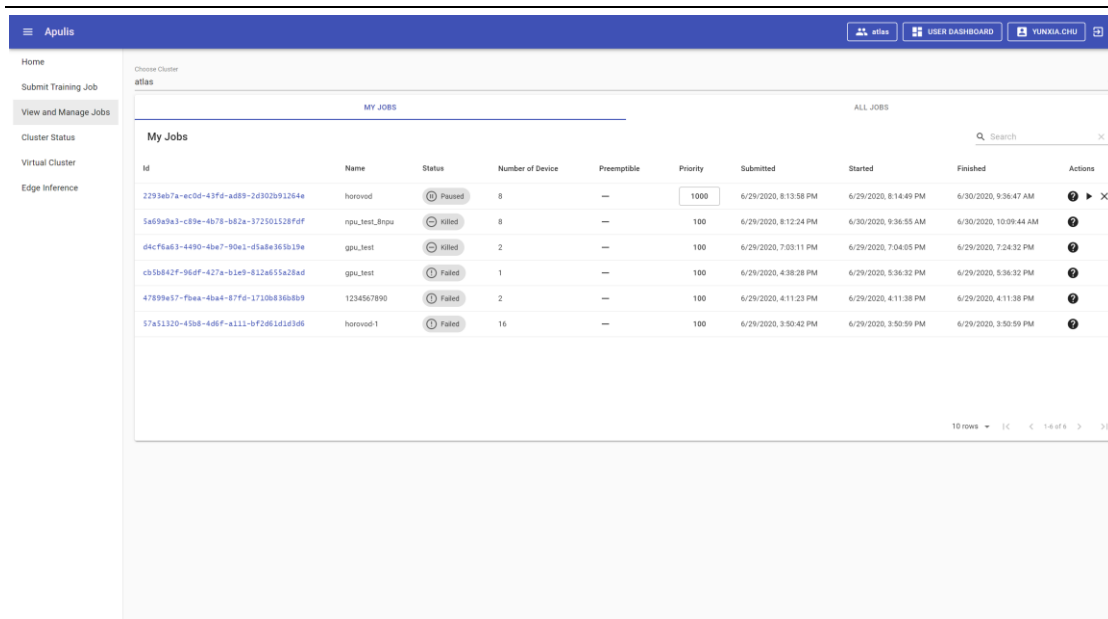
图 8 快捷跳转链接

The screenshot shows the 'Team Virtual Cluster Status' page in the Apulis dashboard. The sidebar menu is the same as in Figure 8. The main content area has a tabbed interface with four tabs: 'TEAM VIRTUAL CLUSTER STATUS' (selected), 'TEAM VC USER STATUS', 'CLUSTER USAGE', and 'PHYSICAL CLUSTER NODE STATUS'. The 'Team Virtual Cluster Status' tab displays a table with the following data:

Name	Device Type	Number of Device	Unschedulable	Used	Preemptible Used	Available	Active Jobs
atlas	huawei_npu_arm64	16	0	16	0	0	1
atlas	nvidia_gpu_amd64	16	0	16	0	0	6

At the bottom right of the table, there is a pagination control showing '< > 1/2 of 2 >'.

图 9 TEAM VIRTUAL CLUSTER STATUS 页签



ID	Name	Status	Number of Device	Preemptible	Priority	Submitted	Started	Finished	Actions
2293eb7a-ec0d-43fd-ad89-2d902b91264e	horovod	Paused	8	—	1000	6/29/2020, 8:13:58 PM	6/29/2020, 8:14:49 PM	6/30/2020, 9:35:47 AM	ⓘ ▶ ✕
5a69a9a3-c89e-4b78-b82a-3721015128fd	npa_test_Bnpa	Killed	8	—	100	6/29/2020, 8:12:24 PM	6/30/2020, 9:36:55 AM	6/30/2020, 10:09:44 AM	ⓘ ?
d4cf6a63-4490-4be7-90e1-d5a8e365b19e	gpu_test	Killed	2	—	100	6/29/2020, 7:03:11 PM	6/29/2020, 7:04:05 PM	6/29/2020, 7:24:32 PM	ⓘ ?
cb5b842f-96df-427a-b1e9-812a655a28ad	gpu_test	Failed	1	—	100	6/29/2020, 4:38:28 PM	6/29/2020, 5:36:32 PM	6/29/2020, 5:36:32 PM	ⓘ ?
47899a57-fbaa-4ba4-87fd-1710b836b8b9	1234567890	Failed	2	—	100	6/29/2020, 4:11:23 PM	6/29/2020, 4:11:38 PM	6/29/2020, 4:11:38 PM	ⓘ ?
57a11320-45b6-4a6f-a111-bf2a61d1d3d6	horovod-1	Failed	16	—	100	6/29/2020, 3:50:42 PM	6/29/2020, 3:50:59 PM	6/29/2020, 3:50:59 PM	ⓘ ?

图 10 MY JOBS 页签

### 3.3. Submit Training Job

点击菜单栏的“Submit Training Job”，可以进入提交 job 页面，如图 11 所示；提交 job 页面包括基本设置，高级设置和模板设置，提交 job 后会跳转到 job 详情页面。

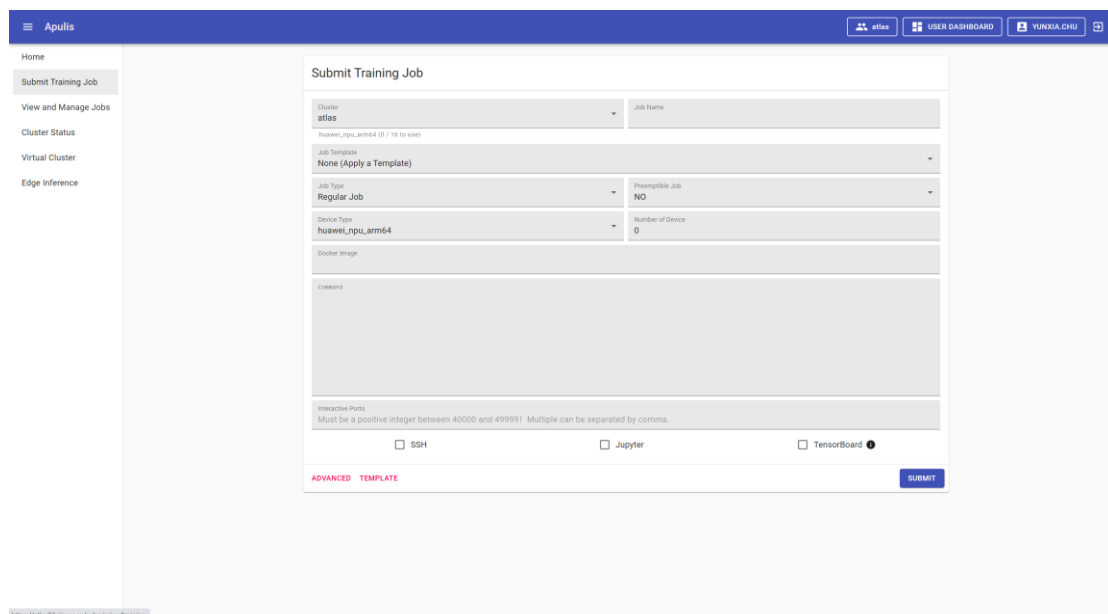


图 5 提交 job 页面

#### 3.3.1 基本设置

基本设置包括 Cluster、Job Name、Job Template、Job Type、Preemptible Job、Device Type、Number of Device、Docker Image、Command、Interactive Ports、

endpoints 共 11 项。其中 Device Type 参数、Number of Device 参数，只能选择所在 VC 资源组中的 AI 计算芯片和数量。

Cluster：物理集群名称，目前只支持 1 个物理集群。

Job Name：必填项，表示训练任务名称。

Job Template：可选择已有的训练任务模板，如图 12。

Submit Training Job

Cluster: sandbox02-gpu01

Job Name:

nvidia (0 / 1 to use)

None (Apply a Template)

- cpu\_test(user)
- cpu-test(team)
- gpu-test(team)
- nni-template(user)
- private-docker-hub(user)
- pytorch-gpu-demo(user)
- test01(user)

Interactive Ports  
Must be a positive integer between 40000 and 49999! Multiple can be separated by comma.

☐ SSH ☐ jupyter ☐ Tensorboard ⓘ

ADVANCED TEMPLATE

SUBMIT

图 6 Template 列表

Job Type：可设置任务类型，分为 Regular Job 和 Distributed Job；Regular Job，表示进行常规训练任务，Distributed Job，表示进行分布式训练任务。

Preemptible Job：该参数用来设置任务是否可被抢占资源；设置为 NO，则表示不可被抢占；设置为 YES，则表示可被抢占。设置为 NO 的 Job 可以抢占设置为 YES 的 Job, 设置为 YES 的 Job 不可以抢占任何的 Job。

Device Type：资源类型，支持华为 huawei\_npu\_arm64 和英伟达 nvidia\_gup\_amd64。

Number of Device：当 Job Type 为 Regular Job 时，才显示这个参数项，表示任务训练时需要的 AI 计算芯片数量，但是不可以超过当前 VC 中的所有 AI 计算芯片的总数量。如果 Device Type 为华为 huawei\_npu\_arm64，那么 Number of Device 只能填入 0、1、2、4、8，这是厂家目前的要求，如图 13。

Submit Training Job

Cluster  
atlas  
huawei\_npu\_arm64 (0 / 16 to use)

Job Name

Job Template  
None (Apply a Template)

Job Type  
Regular Job

Preemptible Job  
NO

Device Type  
huawei\_npu\_arm64

Number of Device  
3  
Must be a positive integer from 0 to 16, and can only be one of 0, 1, 2, 4, 8

Docker Image

Command

Interactive Ports  
Must be a positive integer between 40000 and 49999! Multiple can be separated by comma.

☐ SSH
☐ Jupyter
☐ TensorBoard ⓘ

ADVANCED TEMPLATE

SUBMIT

图 13 Number of device 的输入限制

Number of Nodes: 当 Job Type 为 Distributed Job, 才显示这个参数项, 表示任务训练时需要物理节点的数量, 如图 14。

Submit Training Job

Cluster  
atlas  
huawei\_npu\_arm64 (0 / 16 to use)

Job Name

Job Template  
None (Apply a Template)

Job Type  
Distributed Job

Preemptible Job  
NO

Device Type  
huawei\_npu\_arm64

Number of Nodes  
0

Total Number of Device  
0

Docker Image

Command

Interactive Ports  
Must be a positive integer between 40000 and 49999! Multiple can be separated by comma.

☐ SSH
☐ Jupyter
☐ TensorBoard ⓘ

ADVANCED TEMPLATE

SUBMIT

图 14 Number of Nodes

Total Number of Device: 当 Job Type 为 Distributed Job, 才显示这个参数项, 该参数与 Number of Nodes 联动。目前缺省认为一个物理节点上 AI 计算芯片的数量为 8 个, 即 Number of Nodes 为 1 时, Total Number of Device 为 8; Number of Nodes 为 2 时, Total Number of Device 为 16, 如图 15。

Submit Training Job

Cluster  
atlas  
huawei\_npu\_arm64 (0 / 16 to use)

Job Name

Job Template  
None (Apply a Template)

Job Type  
Distributed Job

Preemptible Job  
NO

Device Type  
huawei\_npu\_arm64

Number of Nodes  
2

Total Number of Device  
16

Dockers Image

Command

Interactive Ports  
Must be a positive integer between 40000 and 49999! Multiple can be separated by comma.

☐ SSH
☐ Jupyter
☐ TensorBoard ⓘ

ADVANCED TEMPLATE

SUBMIT

图 15 Total Number of Device

Docker Image: 必填项, 填写镜像名称, 提交训练任务后会自动下载该镜像。

Command: 必填项, 可以填写启动训练任务的脚本等, 容器启动后会自动执行训练任务; 也可以只填写 sleep infinity, job running 后用户进入容器内手动执行训练任务。

Interactive Ports: 非必填项, 可设置容器内的交互端口, 容器启动后会提供访问链接。【注意】端口设置范围为 40000-49999, 如设置多端口时, 需用英文的逗号分割。

Endpoints: 非必选项, 可设置开启 SSH、Jupyter、Tensorboard, 容器启动后会自动开启 SSH、Jupyter、Tensorboard 服务, 提供访问链接。

### 3.3.2 高级配置

高级配置中包括 Custom Docker Registry, Mount Directories 和 Environment Variables, 见图 16。



☐ SSH☐ jupyter☐ Tensorboard ⓘ

**Custom Docker Registry**

Registry

Username

Password

**Mount Directories**

Path in Container	Path on Host Machine / Storage Server	Enable
/work	<div>Work Path</div>	<input checked="" type="checkbox"/>
/data	<div>Data Path</div>	<input checked="" type="checkbox"/>
/job	<div>Job Path</div>	<input checked="" type="checkbox"/>

**Environment Variables**

Name	Value
<div>+</div>	

ADVANCED TEMPLATE

SUBMIT

图 16 提交 Job 页面-高级配置

Custom Docker Registry：可设置私有 docker 仓库的地址、用户名、密码；【注意】如私有仓库为 http，需修改物理机的/etc/docker/daemon.json 文件后重启 docker 服务。

Mount Directories：用于将用户设置的物理机目录挂载到容器内的对应目录下。Path in Container，表示容器内的目录。Path on Host Machine/Storage Server，表示物理机需要挂载的目录。Enable 开关，如果关闭，则不将用户设置的物理机目录挂载到容器内的目录下；如果打开，则进行挂载，如下描述：

/work 个人工作目录，将平台缺省的物理机路径，/dlwsdata/work/个人用户名，挂载到容器内的/work 目录下。

/data 容器内共享目录，将平台缺省的物理机路径，/dlwsdata/storage，挂载到容器内的/data 目录下。

/job 存放 job 相关文件的目录，将平台缺省的物理机路径，/dlwsdata/jobs/当天日期/jobId，挂载到容器内的/job 目录下。

例如在 Path on Host Machine/Storage Server 的第一行，填入子目录 test，并且 Enable 开关打开，则将物理机路径，/dlwsdata/work/个人用户名/test，挂载到容器内的/work 目录下。如果 test 目录不存在，则会新建目录后再挂载。

Environment Variables：新增环境变量值，容器启动后可以获取和使用。

### 3.3.3 模板设置

模板设置中包括保存模板和删除模板，见图 17。

图 7 提交 job 页面-模板设置

**保存模板：**基本配置和高级配置中的各参数进行设置后，可设置 Template name 保存为模板，方便以后使用。模板保存时可设置 scope，如果 scope 设置为 user，则当前登录用户切换 VC 后都可选择该模板，其他用户则不显示该模板；如果 scope 设置为 team，则其他用户登录后，在当前 VC 中都可选择该模板，切换 VC 后将不显示该模板。

**删除模板：**如果当前 VC 中无模板，则不显示 DELETE 按钮；如当前 VC 中有模板，点击 DELETE 后弹出删除模板页面，选择模板后点击 DELETE 即可删除，见图 18。

图 8 删除模板

## 3.4. View and Manage Jobs

View and Manage Jobs 包括两个页签，分别为 MY JOBS 和 ALL JOBS。MY JOBS 为当前登录用户的所有 job 状态的历史列表；ALL JOBS 为所有用户的 job 列表，只展示排队中、调度中和暂停状态的 job 信息，不展示运行失败和结束的 job 信息，见图 19。

MY JOBS											ALL JOBS										
Running Jobs																					
Id	Name	Status	Number of Device	User	Preemptible	Priority	Submitted	Started	Finished	Actions											
45f9b10e-7d5c-4c51-adc3-b679cf78cd8b	gpu_test	Running	1	wellin	—	100	6/30/2020, 11:50:13 AM	6/30/2020, 12:00:54 PM		🔍    ✕											
391763ad-9b27-4759-a62d-e0a2f2af6859	gpu_test	Running	1	wellin	—	100	6/30/2020, 11:50:03 AM	6/30/2020, 12:00:38 PM		🔍    ✕											
0da5a588-ca99-43dc-9d44-9204320f7abf	gpu_test	Running	2	wellin	—	100	6/30/2020, 11:49:52 AM	6/30/2020, 12:00:31 PM		🔍    ✕											
4393c0c4-2c8f-4695-b2fe-6a59721a2224	gpu_test	Running	4	wellin	—	100	6/30/2020, 11:49:39 AM	6/30/2020, 12:00:20 PM		🔍    ✕											
82c2955c-1d35-4799-bf07-9abc6c842ea8	horovod	Running	8	wellin	—	100	6/30/2020, 11:49:07 AM	6/30/2020, 12:03:47 PM		🔍    ✕											
											5 rows	<	>								
Pauses Jobs																					
Id	Name	Status	Number of Device	User	Preemptible	Priority	Submitted	Started	Finished	Actions											
16a4b39-c3d5-47b8-bde2-689438b7b79c	horovod	Paused	16	wellin	—	100	6/30/2020, 10:55:46 AM	6/30/2020, 11:46:56 AM	6/30/2020, 11:46:56 AM	🔍 ▶ ✕											
77604eb1-2476-4da5-bf04-3ae7f91f4b4c	horovod	Paused	16	bifeng.peng	—	100	6/30/2020, 9:47:19 AM	6/30/2020, 9:48:27 AM	6/30/2020, 10:18:10 AM	🔍 ▶ ✕											
745f8745-fc49-405b-b937-585e53facc37	npu_test_binpu	Paused	16	bifeng.peng	—	100	6/30/2020, 9:35:32 AM	6/30/2020, 10:20:28 AM	6/30/2020, 10:20:28 AM	🔍 ▶ ✕											
2293eb7a-ec0d-43fd-ad89-2d302b91264e	horovod	Paused	8	yunxia.chu	—	1000	6/29/2020, 8:13:58 PM	6/29/2020, 8:14:49 PM	6/29/2020, 9:36:47 AM	🔍 ▶ ✕											
											5 rows	<	>								

图 9 ALL JOBS

### 3.4.1 job 列表

Job 列表中包括十项内容：Id、Name、Status、Number of Device、Preemptible、Priority、Submitted、Started、Finished、Actions，见图 20。

Id：表示 job Id，点击后可跳转到 job 详情页面。

Name：表示 job 名称。

Status：表示 job 的运行状态，运行状态有 queued、scheduling、running、pasuing、paused、finished、error、failed 等。

Number of Device：表示 job 使用的 AI 计算芯片的数量。

Preemptible：表示该 job 的资源是否可被抢占。

Priority：优先级，默认为 100，可设置的范围为 1-1000。

Submitted：显示 job 的提交时间。

Started：显示 job 的运行开始时间。

Finished：显示 job 的运行结束时间。

Actions：可以对 job 进行 support、pause、resume 和 kill 操作，点击 support 按钮时弹出邮件发送页面，可向运维人员发送邮件寻求帮助。

MY JOBS										
Id	Name	Status	Number of Device	Preemptible	Priority	Submitted	Started	Finished	Actions	
b7a61333-d636-4a27-8709-2e13507d0c4c	cpu_test	Scheduling	0	—	100	6/30/2020, 1:11:33 PM				🔍 ⏸ ✕
2293ab7a-ec0d-43fd-ad89-26302b91264e	horovod	Paused	8	—	1000	6/29/2020, 8:13:58 PM	6/29/2020, 8:14:49 PM	6/30/2020, 9:36:47 AM		🔍 ▶ ✕
5a69a9a3-c89e-4b78-b82a-372501528fef	npn_test_8npn	Killed	8	—	100	6/29/2020, 8:12:24 PM	6/30/2020, 9:36:55 AM	6/30/2020, 10:09:44 AM		🔍
d4cf6a63-4490-4be7-90e1-d5a8a365b19e	gpu_test	Killed	2	—	100	6/29/2020, 7:03:11 PM	6/29/2020, 7:04:05 PM	6/29/2020, 7:24:32 PM		🔍
cb5b842f-96df-427a-b1a9-812a655a28ad	gpu_test	Failed	1	—	100	6/29/2020, 4:38:28 PM	6/29/2020, 5:36:32 PM	6/29/2020, 5:36:32 PM		🔍
47899e57-fbea-4ba4-87fd-1710b836b8b9	1234567890	Failed	2	—	100	6/29/2020, 4:11:23 PM	6/29/2020, 4:11:38 PM	6/29/2020, 4:11:38 PM		🔍
57a51320-45b8-4d6f-a111-bf2d61d3d3d6	horovod-1	Failed	16	—	100	6/29/2020, 3:50:42 PM	6/29/2020, 3:50:59 PM	6/29/2020, 3:50:59 PM		🔍

图 20 MY JOBS

### 3.4.2 job 详情

在 MY JOBS 页面或者 ALL JOBS 页面点击 job Id 都可跳转到 job 详情页面，job 详情页包括 4 个页签，分别为 BRIEF、ENDPOINTS、METRICS、CONSOLE，见图 21。

npn_test_8npn			
BRIEF	ENDPOINTS	METRICS	CONSOLE
<p>Job Id 6276665d-9b8e-4bcb-b4e2-6c9f41bbcb5e</p> <p>Job Name npn_test_8npn</p> <p>VcName atlas</p> <p>Docker Image apulisai/mindspore-0.3.0-withtools</p> <p>Command sudo -E bash -c 'cd /data/resnet50_cifar10/ &amp;&amp; mkdir -p /var/log/npn/conf/rlog/ &amp;&amp; cp slog.conf /var/log/npn/conf/rlog/ &amp;&amp; ./run_d.sh &amp;&amp; sleep infinity'</p> <p>Data Path /dwsdata/storage/</p> <p>Work Path /dwsdata/work/bifeng.peng</p> <p>Job Path /dwsdata/work/bifeng.peng/jobs/200630/6276665d-9b8e-4bcb-b4e2-6c9f41bbcb5e</p> <p>Preemptible X</p> <p>Number of Nodes 2</p> <p>Total of Device 16</p> <p>Job Status Running</p> <p>Job Submission Time 6/30/2020, 10:26:49 AM</p>			

图 21 job 详情-BRIEF 页面

BRIEF: Job 简介页面，包括 Job id、Job Name、VcName、docker Image、Command、Data Path、Work Path、Job Path、Preemptible、Device Type、Number of Device（Number of Nodes、Total of Device）、Job Status、Job Submission Time。

ENDPOINTS: 如提交 job 时已启用 SSH、Jupyter、Tensorboard, 已设置 Interactive Port, 当 job 在 running 时会显示访问地址, 见图 22, Jupyter、Tensorboard、Interactive Port 点击后即可跳转访问; 如提交 job 时未启用 SSH、Jupyter、Tensorboard, 在 job 运行结束前都可启用。也可设置新的 Interactive Port。【注意】配置 interactive port 后需在容器内部启用该端口才可访问, 训练任务的日志需存放在指定目录 tensorboard 才可正确访问。

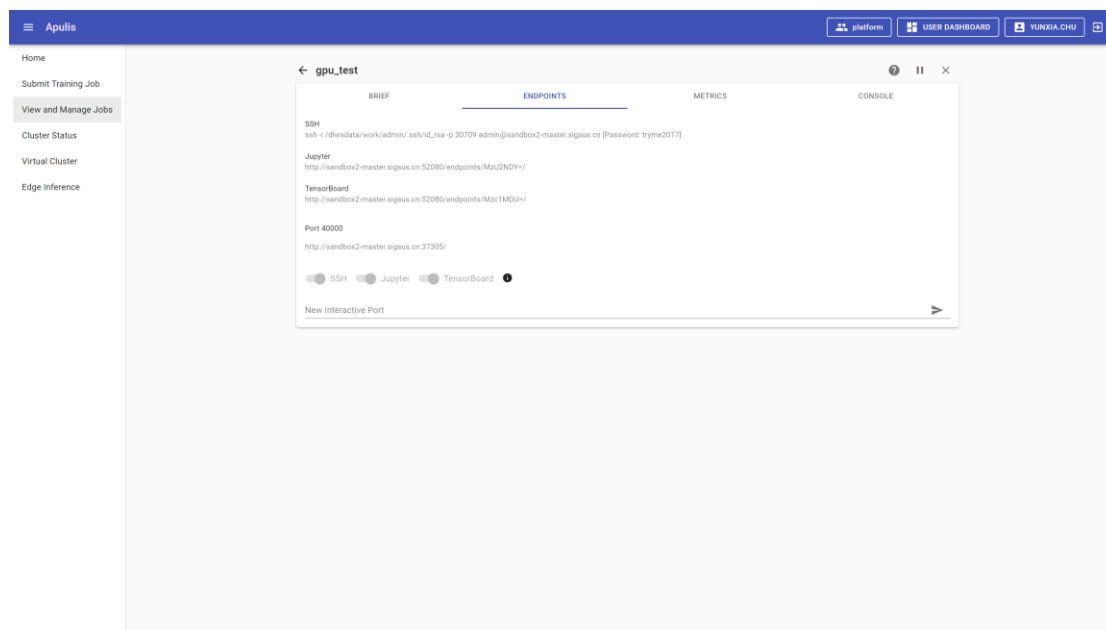


图 10 job 详情-ENDPOINTS 页面

METRICS: 通过 grafana 监控 job 的资源占用, 包括 CPU、Memory Usage、Network、Block IO、GPU Usage、GPU Memory Usage、NPU Utilization、NPU Memory, 见图 23。

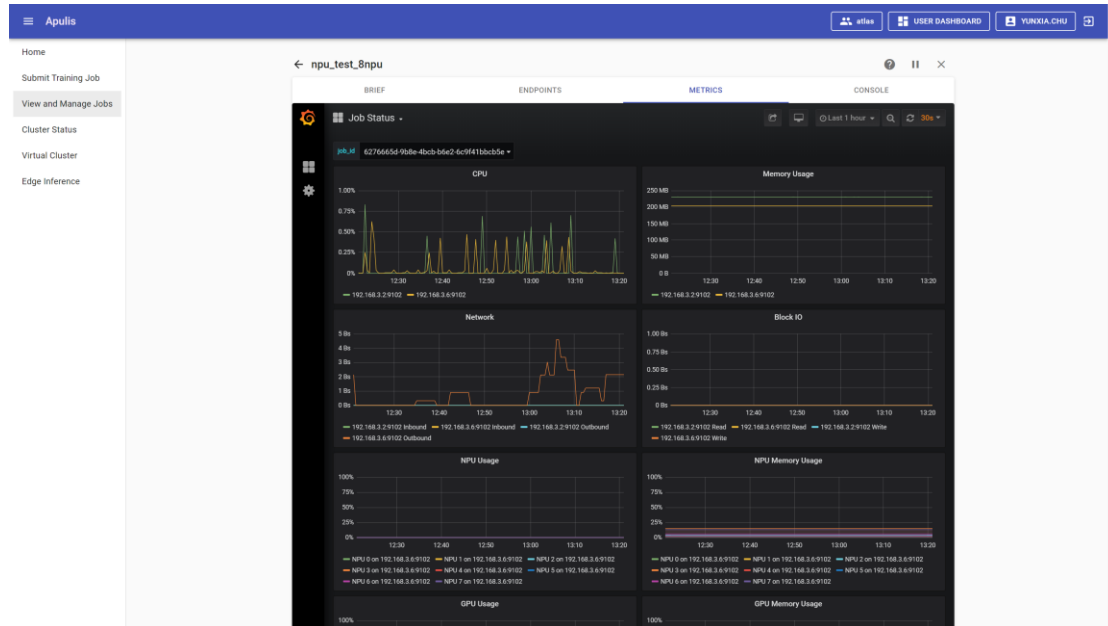


图 11 job 详情-METRICS 页面

CONSOLE: job 运行时的日志输出，显示最新的 2000 行，见图 24。

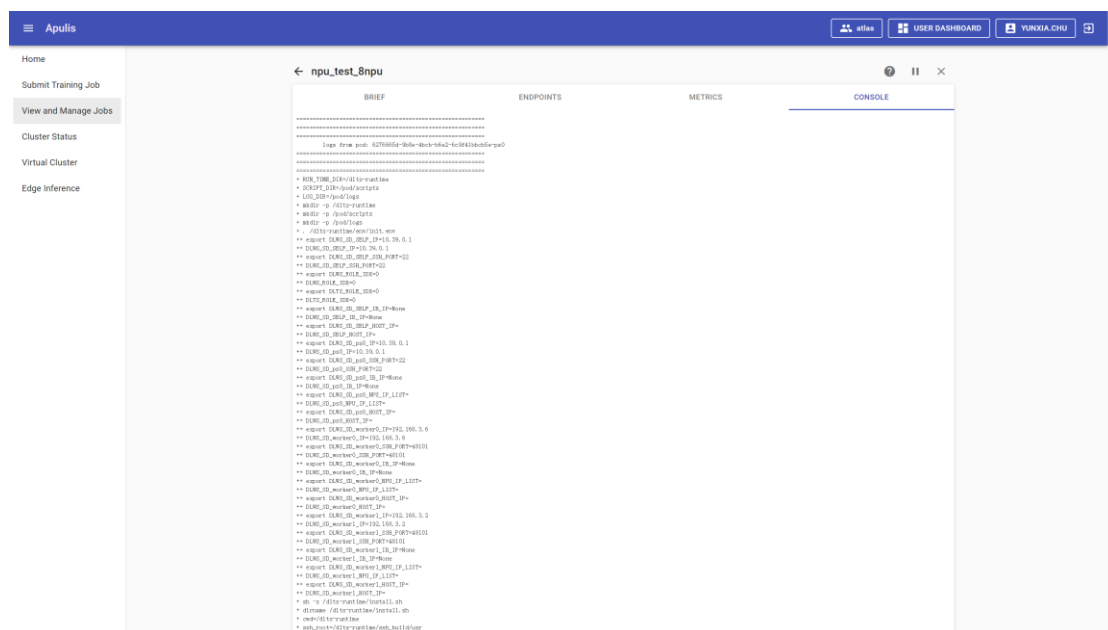


图 12 job 详情-CONSOLE 页面

### 3.5. Cluster Status

Cluster Status 菜单包括 Team Virtual Cluster Status、Team VC User Status、Cluster Usage、Physical Cluster Node Status 共 4 个页签，见附图 25。



Name	Device Type	Number of Device	Unschedulable	Used	Preemptible Used	Available	Active Jobs
atlas	huawei_npu_arm64	16	0	16	0	0	2
atlas	nvidia_gpu_arm64	16	0	16	0	0	5

图 13 Cluster Status

### 3.5.1 Team Virtual Cluster Status

该页签显示虚拟集群的状态统计列表，见上方附图 25，内容包括：

Name：显示物理集群的名称；

Device Type：显示 AI 计算芯片类型；

Number of Device：显示全部 AI 计算芯片的数量；

Unschedulable：显示不可调度 AI 计算芯片的数量；

Used：显示使用中的 AI 计算芯片的数量；

Preemptible Used：显示使用中的可被抢占的 AI 计算芯片的数量；

Available：显示可用的 AI 计算芯片的数量；

Active Jobs：显示运行中的 job 数量。

### 3.5.2 Team VC User Status

默认展示使用当前 VC 资源的用户列表，见附图 26；可切换显示全部用户列表，见附图 27；列表内容包括 Username、Device Type、Current Allocated、Current Allocated Preemptible、Currently Idle、Past Month Booked Hour、Past Month Idle Hour、Past Month Idle Hour%共八项。

Apulis

Home

Submit Training Job

View and Manage Jobs

Cluster Status

Virtual Cluster

Edge Inference

atlas

USER DASHBOARD

YUNKIA.CHU

TEAM VIRTUAL CLUSTER STATUS

TEAM VC USER STATUS

CLUSTER USAGE

PHYSICAL CLUSTER NODE STATUS

Team VC User Status

Search

Username	Device Type	Currently Allocated	Currently Allocated Preemptible	Currently Idle	Past Month Booked Hour	Past Month Idle Hour	Past Month Idle Hour %
bifeng peng	huawei_npu_arm64	16	0	0	12	12	100
wellin	nvidia_gpu_arm64	16	0	0	28	19	67

图 14 当前使用 GPU/NPU 资源的用户列表

Apulis

Atlas

User Dashboard

YUNXIA.CHU

Home

Submit Training Job

View and Manage Jobs

Cluster Status

Virtual Cluster

Edge Inference

TEAM VIRTUAL CLUSTER STATUS

TEAM VC USER STATUS

CLUSTER USAGE

PHYSICAL CLUSTER NODE STATUS

Team VC User Status

Search

X

Username	Device Type	Currently Allocated	Currently Allocated Preemptible	Currently Idle	Past Month Booked Hour	Past Month Idle Hour	Past Month Idle Hour %
bifeng peng	huawei_npu_arm64	16	0	0	12	12	100
wellin	nvidia_gpu_arm64	16	0	0	28	19	67
bifeng peng	nvidia_gpu_arm64	0	0	0	17	14	82
yunxia chu	nvidia_gpu_arm64	0	0	0	108	106	98
yunxia chu	huawei_npu_arm64	0	0	0	0	0	-
Total	huawei_npu_arm64,nvidia_gpu_arm64	32	0	0	165	151	91

图 15 所有使用 VC 资源的用户列表

Username: 显示用户名。

Device Type: 显示 AI 计算芯片的类型。

Current Allocated: 显示当前 AI 计算芯片的分配数量。

Current Allocated Preemptible: 显示当前分配的可被抢占的 AI 计算芯片的数量。

Currently Idle: 显示当前 AI 计算芯片的空闲数量。

Past Month Booked Hour: 显示过去一个月，AI 计算芯片被占用的小时数。

Past Month Idle Hour: 显示过去一个月，AI 计算芯片被占用但利用率为 0 的小时数。

Past Month Idle Hour%: 显示过去一个月，AI 计算芯片被占用但利用率为 0 的时间占比。

### 3.5.3 Cluster Usage

包括 VC Device Usage 和 Cluster Usage 两个监控页面，见图 28。

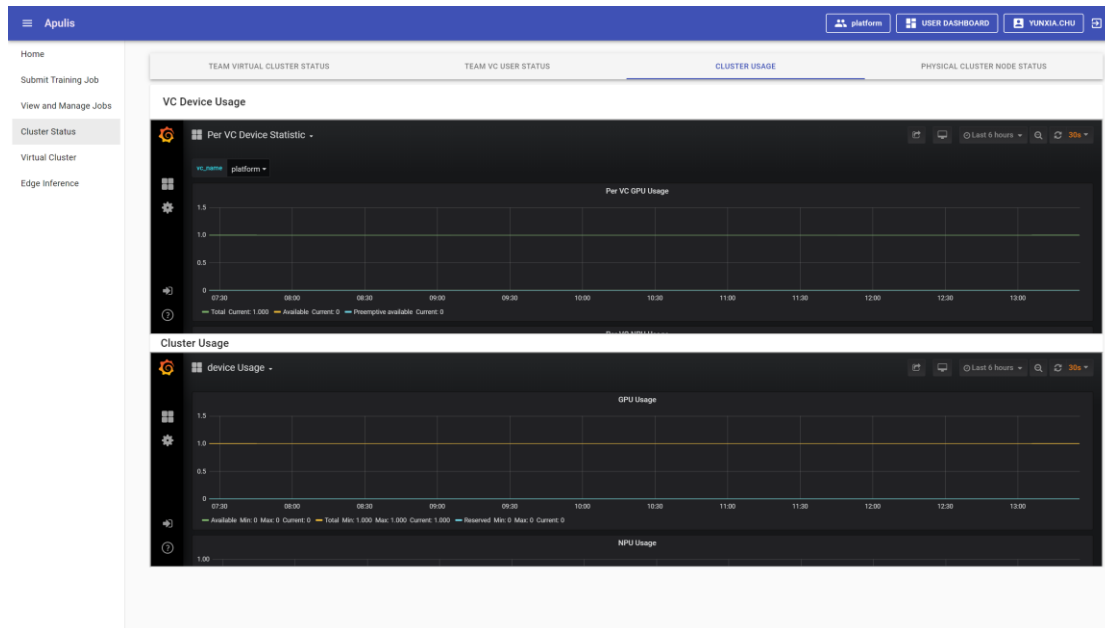


图 16 cluster usage

### 3.5.4 Physical Cluster Node Status

显示物理集群的节点状态，见附图 29；包括 Node Name、Node IP、Device Type、Number of Device、Used、Preemptible Used、Available、Staus、Pods 共 9 项。

Home	TEAM VIRTUAL CLUSTER STATUS	TEAM VC USER STATUS	CLUSTER USAGE	PHYSICAL CLUSTER NODE STATUS
Submit Training Job				
View and Manage Jobs				
Cluster Status				
Virtual Cluster				
Edge Inference				

Node Name	Node IP	Device Type	Number of Device	Used	Preemptible Used	Available	Status	Pods
atlas02	192.168.3.2	huawei_npu_arm64	8	8	0	0	✓	[6276665d-9b8e-4bcb-b6e2-4c9f41bbcb5e-worker1 : bifeng.peng (NPU #8)] [b7a61333-d636-4a27-8709-2a1350790c4c : yunxia.chu (NPU #0)] [custom-user-dashboard-backend-arm64-md5f5 (gpu #0)] [custom-user-dashboard-frontend-arm64-zbbkz (gpu #0)] [jobmanager-arm64-d8qp (gpu #0) (gpu #0)] [nginx-arm64-nfcdh (gpu #0)] [openmst-arm64-4388h (gpu #0)] [prometheus-operator-669669965d-wcfr (gpu #0)] [replicamanager2-arm64-9fbb6 (gpu #0)] [testfapi-arm64-p9fmg (gpu #0)] [webui3-arm64-5bvfkm (gpu #0)]
atlas01	192.168.3.6	huawei_npu_arm64	8	8	0	0	✓	[6276665d-9b8e-4bcb-b6e2-4c9f41bbcb5e-ps0 : bifeng.peng (NPU #0)] [6276665d-9b8e-4bcb-b6e2-4c9f41bbcb5e-worker0 : bifeng.peng (NPU #8)] [nginx-arm64-5dqpz (gpu #0)]
atlas-gpu02	192.168.3.3	nvidia_gpu_arm64	8	8	0	0	✓	[0da5a588-ca99-43dc-9d44-92043207abf : wellin (gpu #2)] [91763ad-9b27-4739-a62d-4ba2af6859 : wellin (gpu #1)] [439160c4-2c0f-4d95-5d2e-6a59721e2224 : wellin (gpu #4)] [46f8b15e-7d8c-4c61-a6c3-bd79d78c0bb : wellin (gpu #1)] [nginx-ix74 (gpu #0)]
atlas-gpu01	192.168.3.4	nvidia_gpu_arm64	8	8	0	0	✓	[82c2955c-1d35-4799-bf07-9abc0c842ea8-ps0 : wellin (gpu #0)] [82c2955c-1d35-4799-bf07-9abc0c842ea8-worker0 : wellin (gpu #8)] [nginx-wx4c4 (gpu #0)]

图 17 physical cluster node status

Node Name：显示物理集群中的各节点名称。

Node IP：显示物理集群中各节点的 IP 地址。

Device Type：显示各节点的 AI 计算芯片类型。

Number of Device：显示各节点的 AI 计算芯片数量。

Used：显示各节点中正在使用的 AI 计算芯片数量。

Preemptible Used：显示各节点中正在使用的可被抢占的 AI 计算芯片数量。

Available：显示各节点中可用的 AI 计算芯片数量。

Status：显示节点状态。

Pods：显示节点上运行的 pods。

## 3.6. Virtual Cluster

显示已配置的虚拟集群列表，见附图 30。

VcName	quota	permissions	actions
atlas	1huawei_npu_arm64*16;1nvidia_gpu_arm64*16	Admin	MODIFY DELETE
test	1huawei_npu_arm64*0;1nvidia_gpu_arm64*0	Admin	MODIFY DELETE

图 30 virtual cluster 列表

### 3.6.1 VC 列表

列表内容包括 VcName、quota、permissions 和 acitons 共 4 项。

VcName：显示 VC 名称。

quota：显示 AI 计算芯片的类型和数量。

Permissions：显示权限。

Acitons：显示操作列，包括修改和删除按钮。

### 3.6.2 新增 VC

点击列表上方的 ADD，打开新增 VC 页面，见附图 31，输入 VcName，选择 Type，设置 Value 参数后保存，即可新增 VC。VcName 不能与已有的名称重复，Value 设置时不可超过剩余可用的 AI 计算芯片的数量。

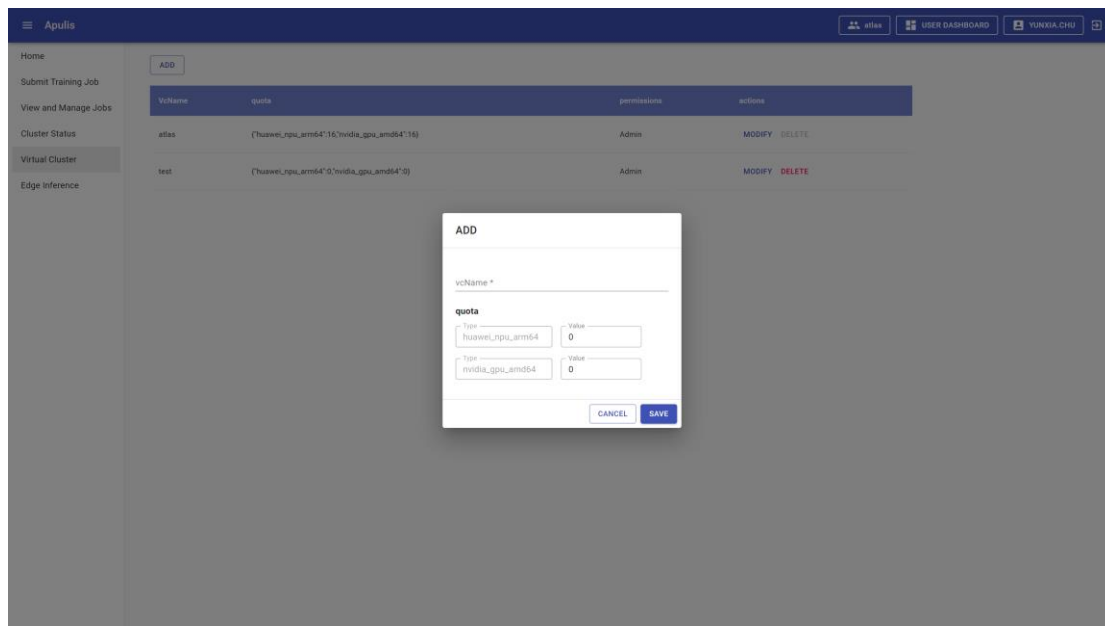


图 18 新增 VC

### 3.6.3 修改 VC

选择 VC，点击列表中的 MODIFY，打开修改 VC 页面，见附图 32；修改 VC 时只能修改 value 参数，VcName 不能编辑修改。【注意】如选择的 VC 有 running、scheduling、killing、pausing 状态的 job，则无法修改 VC。

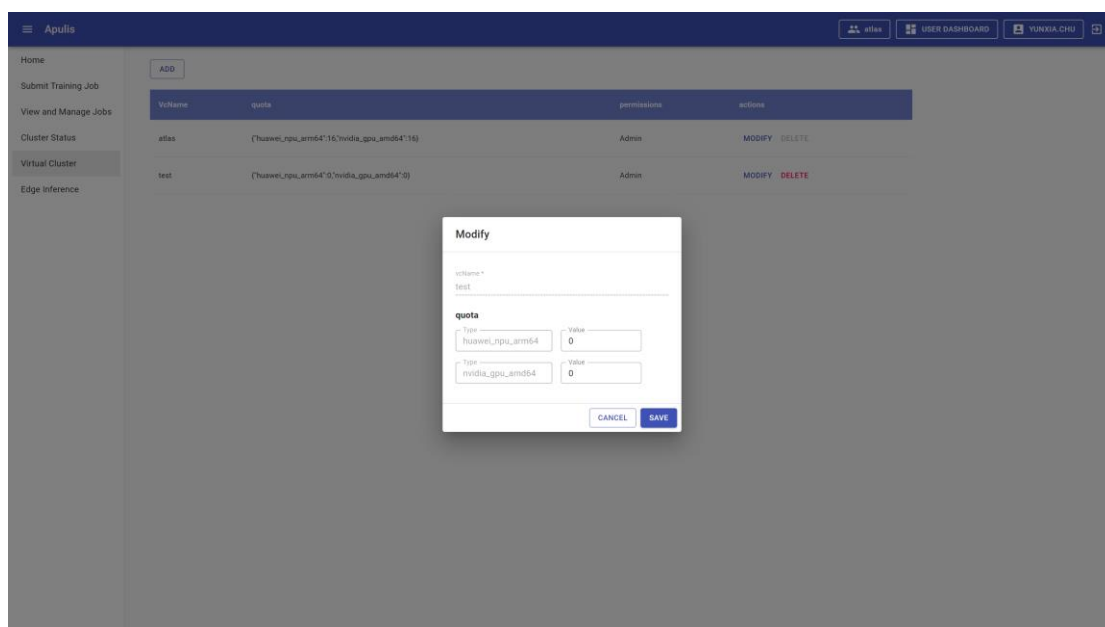


图 19 修改 VC



### 3.6.4 删除 VC

选择 VC，点击删除，弹出删除确认页面，见附图 33，点击 delete 后即删除 VC，同时会删除该 VC 下的所有 job。【注意】如选择的 VC 有 running、scheduling、killing、pausing 状态的 job，则无法删除；当前所在的 VC 删除按钮置灰，无法删除。

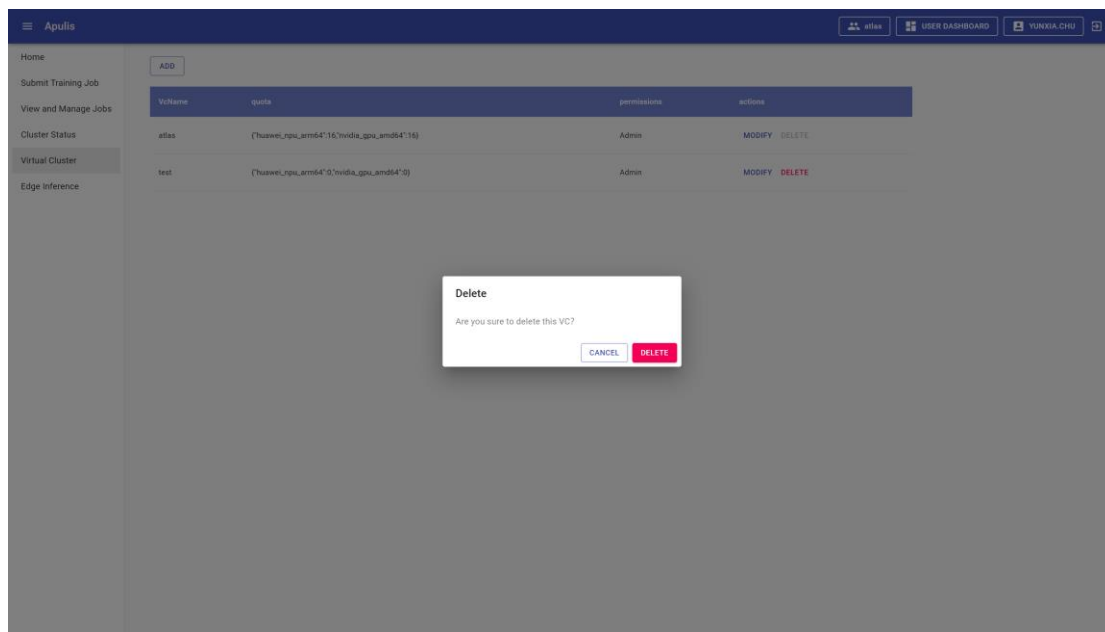


图 20 删除 VC

## 3.7. USER DASHBOARD

User Dashboard 页面如图 34 所示，包括上方菜单栏、dashboard 和 Admin，其中 Admin 包括 User、Group、Role。

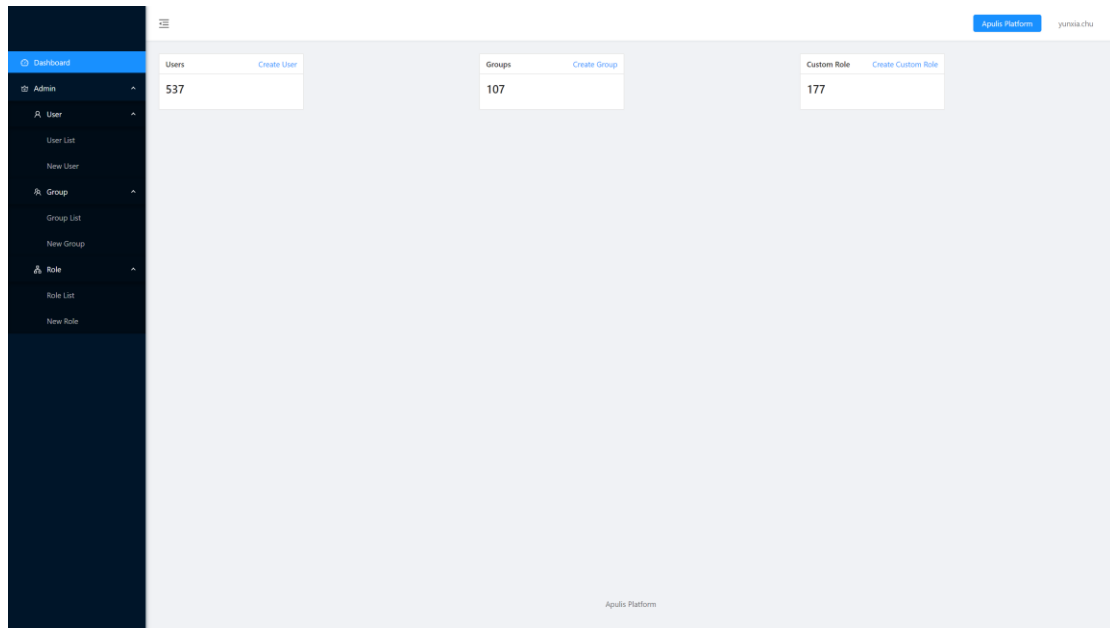


图 21 User Dashboard

### 3.7.1 顶端菜单栏

顶端菜单栏包括 Apulis Platform 和用户名。

点击 Apulis Platform 后切换到深度学习平台，点击用户名后可选择 Account info 或者 logout。选择 Account info 后打开用户详情页，见附图 35，选择 logout 后退出登录，跳转到登录页面。

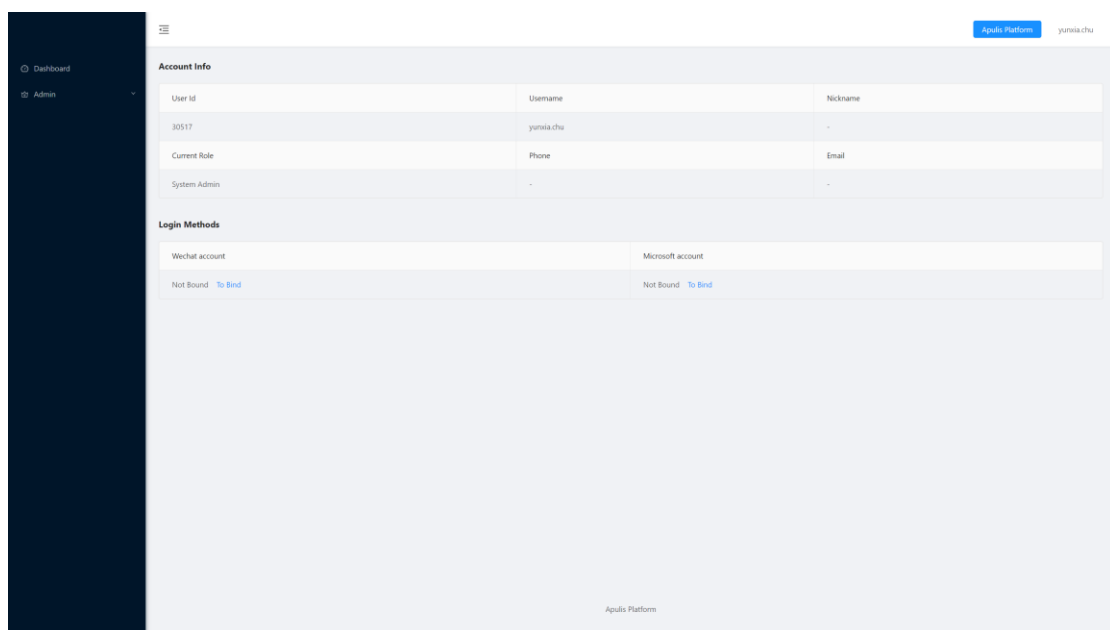


图 22 account info-用户详情页

用户详情页包括 Account Info 和 Login Methods，Account Info 显示用户的基本信息，包括 User Id、Username、Nickname、Current Role、Phone 和 Email，Login Methods 显示是否已绑定 Wetchat 和 Microsoft account，如未绑定，点击 To Bind 后跳转到绑定页面。

### 3.7.2 Dashboard

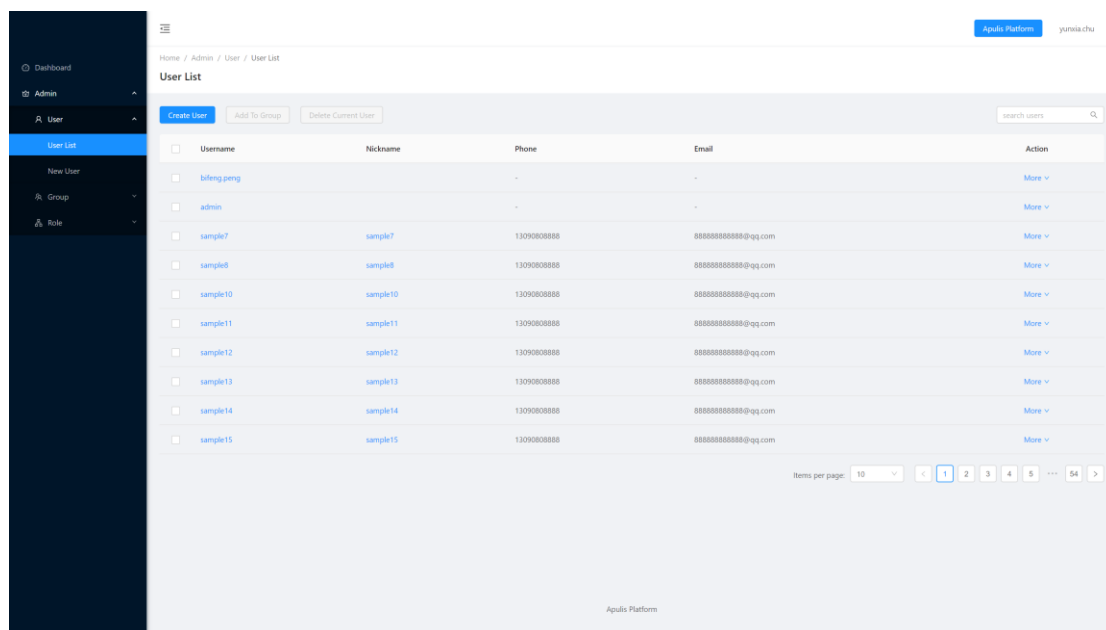
显示已有的用户数、用户组数和角色数，点击 Create User 后即可快捷跳转到新建 User 界面；点击 Create Group 后即可快捷跳转到新建 Group 页面；点击 Create Custom Role 后即可快捷跳转到新建 Role 页面。

### 3.7.3 User

包括 User list 和 New User，User List 展示已有的用户列表，点击 Create User 可创建新的用户。

#### 3.7.3.1 用户列表

User List 中包括 Username、Nickname、Phone、Email、Action 共 5 列，见附图 36。



Username	Nickname	Phone	Email	Action
lifeng.gong		-	-	More ▾
admin		-	-	More ▾
sample7	sample7	1309080888	8888888888@qq.com	More ▾
sample8	sample8	1309080888	8888888888@qq.com	More ▾
sample10	sample10	1309080888	8888888888@qq.com	More ▾
sample11	sample11	1309080888	8888888888@qq.com	More ▾
sample12	sample12	1309080888	8888888888@qq.com	More ▾
sample13	sample13	1309080888	8888888888@qq.com	More ▾
sample14	sample14	1309080888	8888888888@qq.com	More ▾
sample15	sample15	1309080888	8888888888@qq.com	More ▾

图 23 用户列表

Username：用户名，该字段是唯一的，不能重复。

Nickname：用户昵称。

Phone：电话号码。

Email：邮箱。

Action：操作列，包括编辑用户的角色、将用户关联到用户组和删除用户操作，其中拥有系统管理员角色的用户，不能修改此用户的角色，也不能删除此用户。

点击列表中的用户名或昵称可跳转到用户详情页，见附图 37。

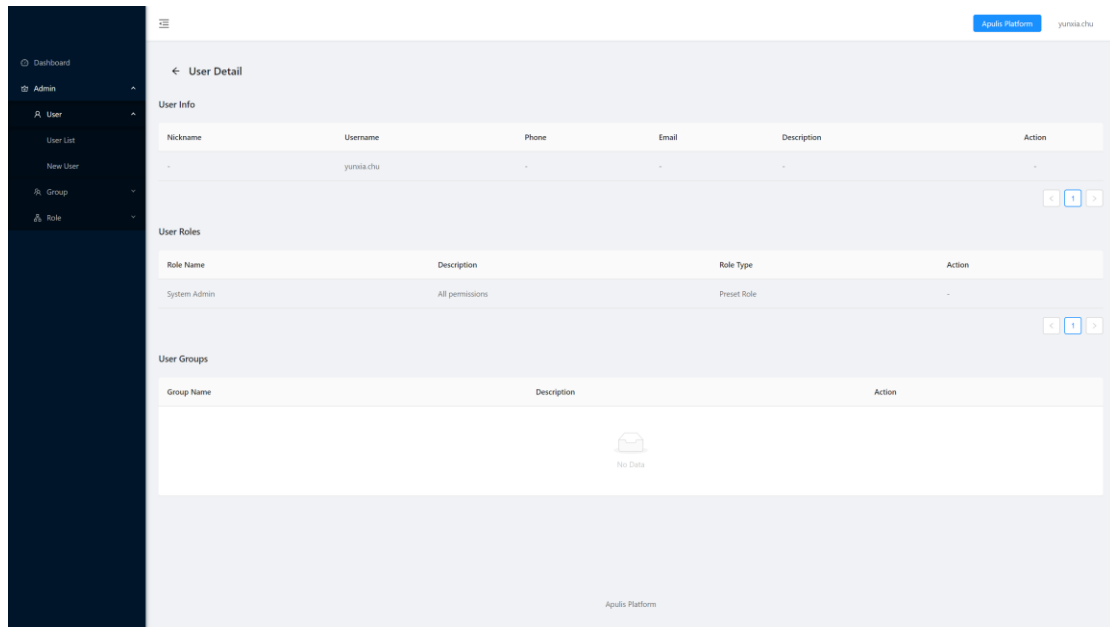


图 24 用户详情页面

### 3.7.3.2 新建用户

创建用户时共分为 3 步，填写用户信息->关联角色->确认，见附图 38-图 40。

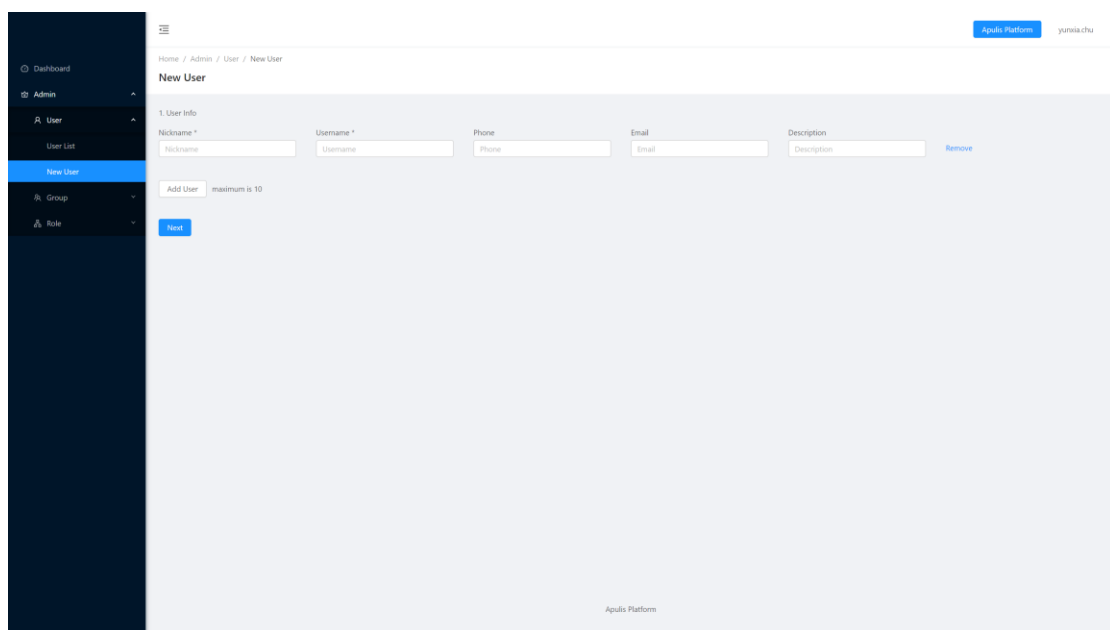


图 38 新建用户-填写用户信息

昵称和用户名必为必填字段，填写之后点击下一步，跳转到关联角色页面。也可以批量添加用户，最大数量不能超过 10 个。

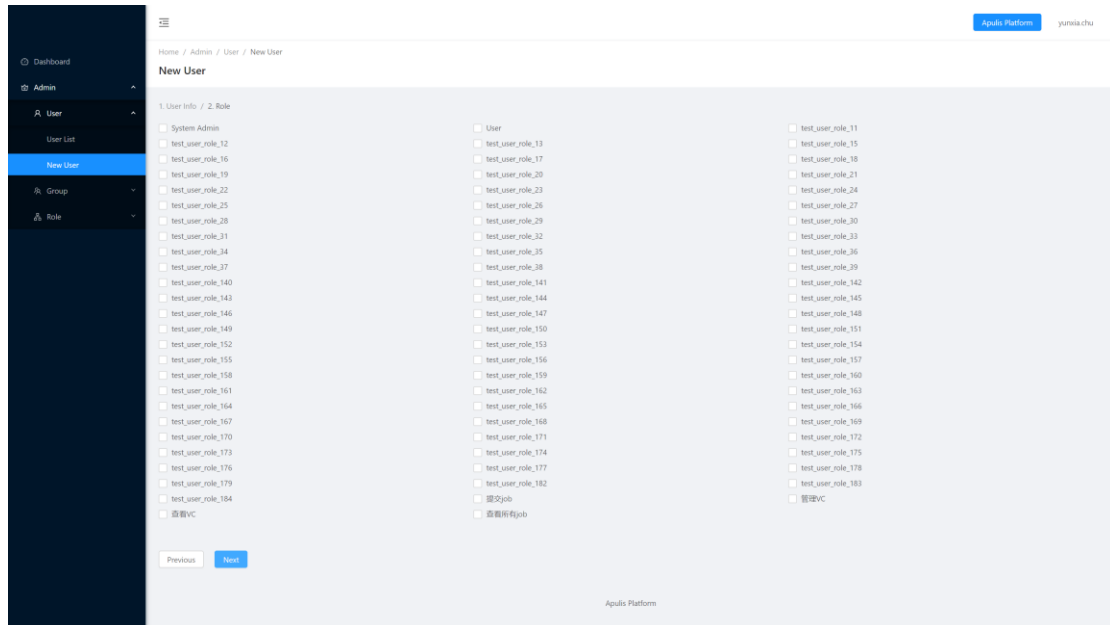


图 39 新建用户-关联角色

关联角色时，至少需选择一个角色关联，点击 next 后跳转到预览页面，也可点击 previous 返回上一步。

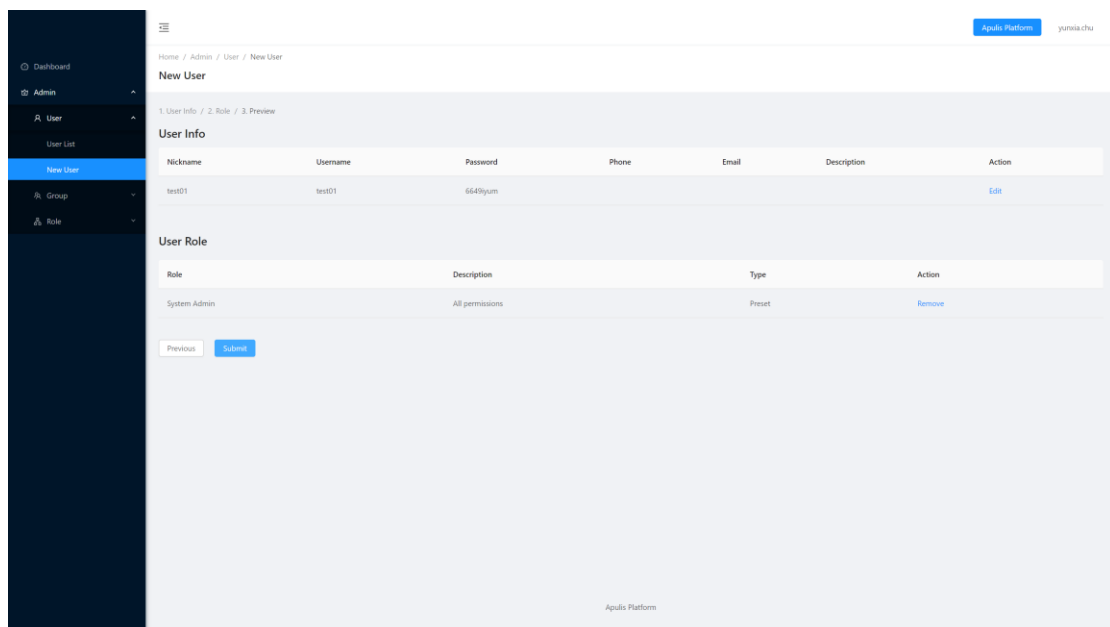


图 40 新建用户-确认

展示新建用户的相关信息，点击 Submit 则新建用户，点击 Previous 则返回上一步。

### 3.7.4 Group

包括 Group List 和 Create Group，Group List 展示已有的用户组列表，点击 Create Group 可创建新的用户组。

#### 3.7.4.1 用户组列表

Group List 中包括 Group Name、Description、Create Time、Action 共 4 列，见附图 41。

Group Name	Description	Create Time	Actions
lin_test	lin_test	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_3	per@id_3 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_4	per@id_4 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_5	per@id_5 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_6	per@id_6 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_7	per@id_7 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_8	per@id_8 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_9	per@id_9 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_10	per@id_10 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>
test_user_group_11	per@id_11 =====	2020-06-19	<a href="#">Add Users</a> <a href="#">Delete</a>

图 25 用户组列表

Group Name：用户组名称，该字段是唯一的，不能重复。

Description：用户组的描述。

Create Time：用户组的创建时间。

Actions：操作列，可以为用户组添加用户和删除用户组。

点击列表中的用户组名称后跳转到用户组详情页面，见附图 42，显示用户组信息、用户组拥有的角色和用户组内的用户。



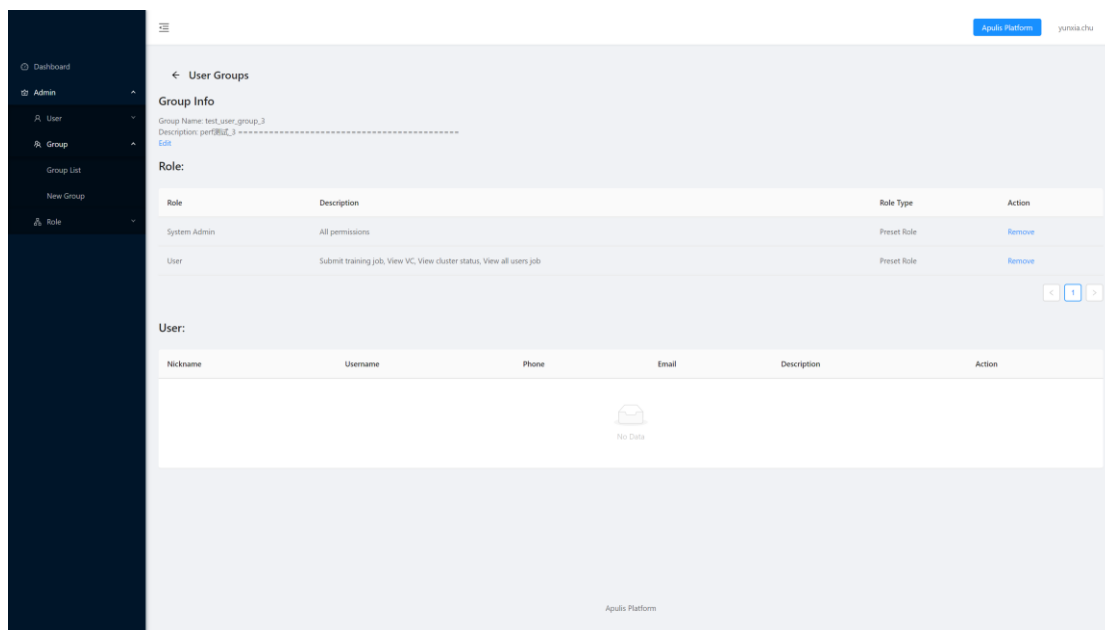


图 26 用户组详情页

### 3.7.4.2 新建用户组

创建用户时共分为 3 步，填写用户组信息->关联角色->确认，见附图 43-图 45。

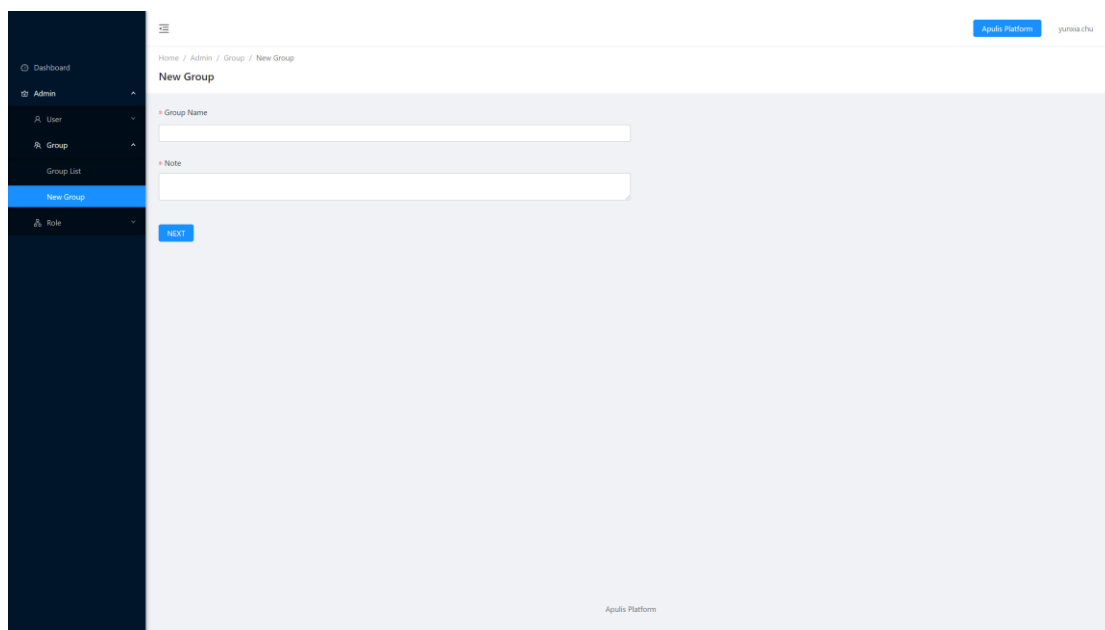


图 27 新建用户组-用户组信息

用户组名称和描述都为必填字段，填写后点击下一步，跳转到关联角色页面。

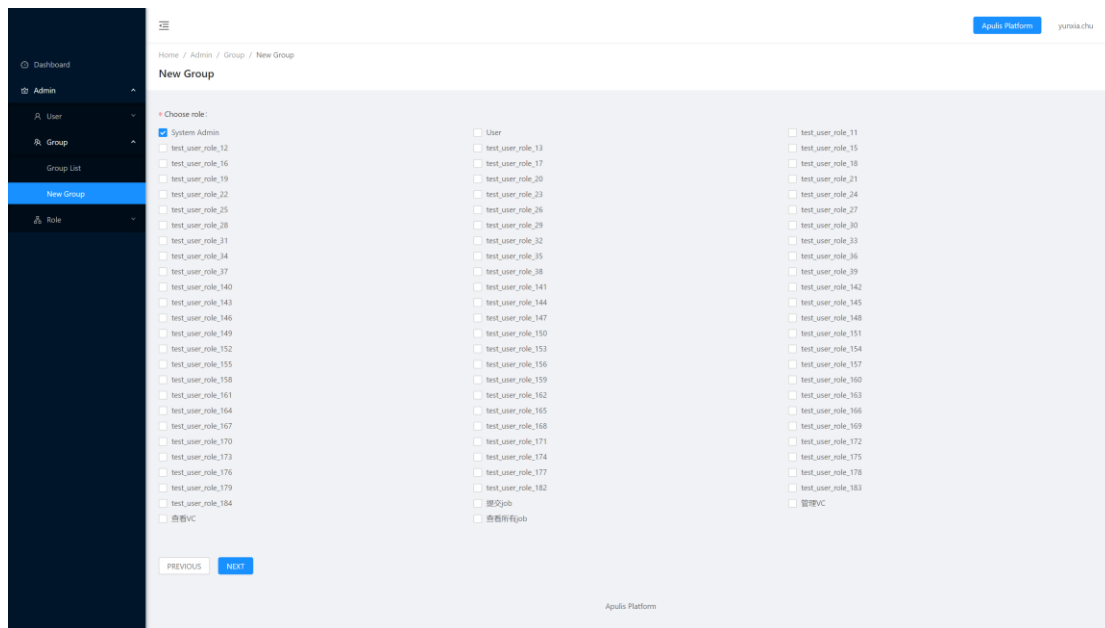


图 284 新建用户组-关联角色

至少需选择一个角色关联，点击 next 后跳转到确认页面，也可点击 previous 返回上一步。

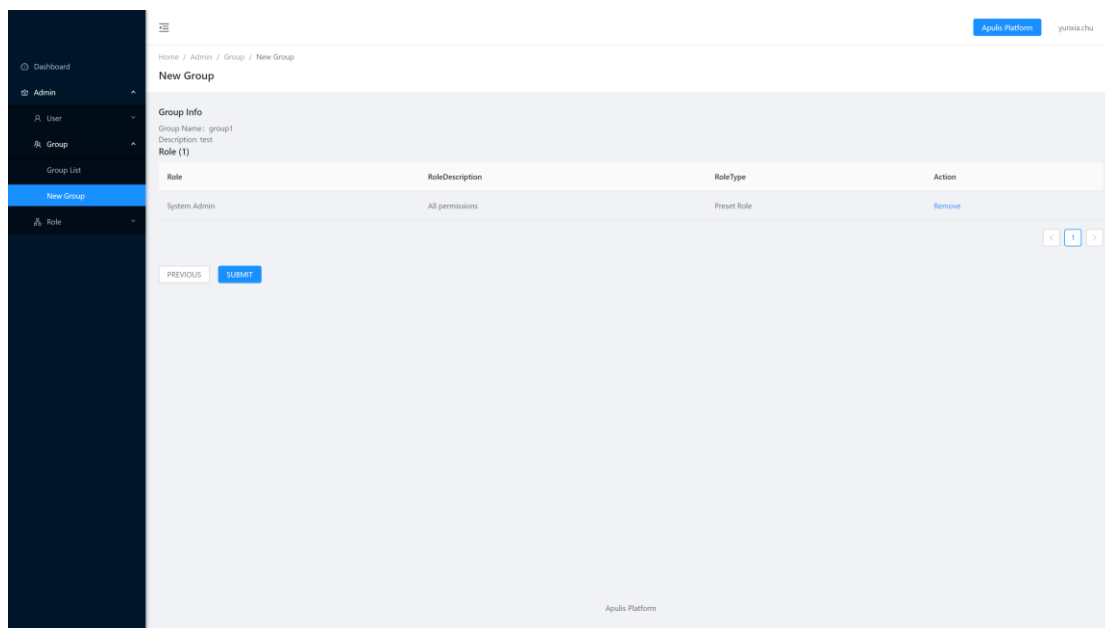


图 295 新建用户组-确认

展示新建的用户组相关信息，点击 submit 新建用户组，点击 previous 返回上一步。

### 3.7.5 Role

包括 Role List 和 Create Role，Role List 展示已有的角色列表，点击 Create role 可创建新的角色。

### 3.7.5.1 角色列表

Role List 中包括 Role Name、Description、Type、Action 共 4 列，见附图 46。

Role Name	Description	Type	Action
<input type="checkbox"/> System Admin	All permissions	Preset Role	<a href="#">Related To User</a> <a href="#">Related To Group</a>
<input type="checkbox"/> User	Submit training job, View VC, View cluster status, View all users job	Preset Role	<a href="#">Related To User</a> <a href="#">Related To Group</a>
<input type="checkbox"/> test_user_role_11	permission_11 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_12	permission_12 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_13	permission_13 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_15	permission_15 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_16	permission_16 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_17	permission_17 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_18	permission_18 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>
<input type="checkbox"/> test_user_role_19	permission_19 ~~~~~	Custom Role	<a href="#">Related To User</a> <a href="#">Related To Group</a> <a href="#">Delete</a>

图 30 角色列表

Role Name：角色名称，该字段是唯一的，不能重复。

Description：角色的描述。

Type：角色的类型，角色类型分为预设角色和用户自定义角色两种。预设角色是系统默认创建的，不能删除。

Actions：操作列，可以进行关联用户、关联用户组和删除角色的操作。

### 3.7.5.2 新建角色

创建角色时共分为 3 步，填写角色名称->描述->选择权限，见附图 47。

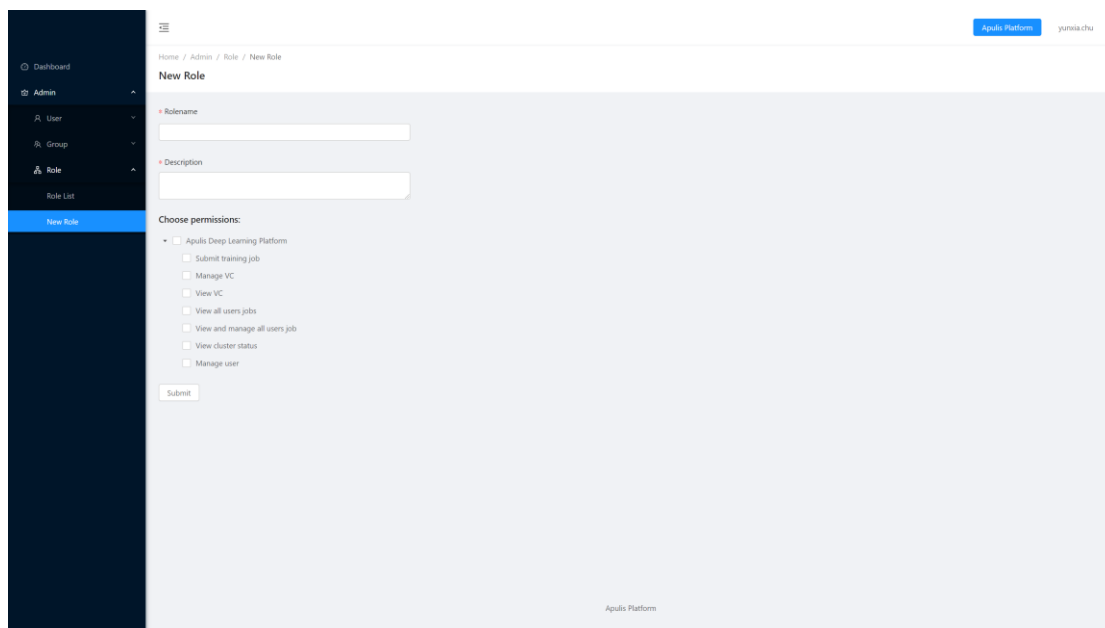


图 31 新建角色

平台的权限共分为 7 种：

Submit training job：提交训练任务。

Manage VC：管理 VC，包括新增、编辑和删除 VC。

View VC：只能查看 VC，不可以新增、编辑和删除 VC。

View all users jobs：查看所有用户的 job，无法对 job 进行操作。

View and manage all users job：查看和管理所有用户的 job，包括对 job 进行 pause、kill 等操作。

View cluster status：查看集群状态。

Manager：用户管理权限。

### 3.8. 报警配置

平台暂无可可视化的报警配置页面（后续会增加），可通过修改配置文件后重新部署 repairmanager2 服务，来使相关报警配置参数生效。修改报警参数后，需执行的命令如下：

重新编译 repairmanager2 服务，如有 x86 架构的机器，需在 x86 机器上添加参数--arch amd64 执行，如有 arm 架构的机器，需在 arm 机器上添加参数--arm arm64 执行：

重新编译：

```
#sudo ./deploy.py --verbose --arch amd64 docker push repairmanager2
```

重启 repairmanager2 服务：

```
#sudo ./deploy.py kubernetes stop repairmanager2
```

```
#sudo ./deploy.py kubernetes start repairmanager2
```

报警参数相关的配置文件共 4 个，分别为：

apulis\_platform/src/ClusterBootstrap/config.yaml，配置发件人和收件人的信息、报警标题中的集群名称；

apulis\_platform/src/RepairManager/config/ecc-config.yaml，配置监控指标的阈值表达式、查询间隔、查询时间段、阈值百分比；

apulis\_platform/src/RepairManager/config/rule-alerts.yaml，配置是否启用报警、是否重复报警和重复报警的时间间隔；

apulis\_platform/src/RepairManager/config/rule-config.yaml，配置查询的时间间隔。

目前监测的报警指标共 4 个，分别为 nvidiasmi\_ecc\_error\_count、cpu utilization、memory utilization 和 filesystem left space。

### 3.8.1 config.yaml

cluster\_name：配置报警邮件标题中的集群名称。

smtp\_url：配置 smtp 服务器地址。

login 和 sender：配置发件人的邮箱。

password：配置发件人邮箱的密码。

receiver：配置收件人邮箱，可配置多个。

```
repair-manager:
  cluster_name: "atlas"
  ecc_rule:
    cordon_dry_run: True
  alert:
    # smtp_url: smtp.office365.com
    # login:
    # smtp_url: smtp.office365.com
    # login:
    # password:
    # sender:
    smtp_url: smtp.qq.com
    login:
    password:
    sender:
    receiver: [" "]
  enable_custom_registry_secrets: True
```

图 48 config.yaml 配置

### 3.8.2 ecc-config.yaml

query\_expression: 配置监控指标的阈值表达式，当监控指标的值达到该条件时记录一次报警；

step: 配置查询间隔，单位为分钟，假设该值设置为 N，则表示每隔 N 分钟去查询一次监控指标的值；

interval: 配置查询的时间段，单位为分钟，假设该值设置为 M，则表示在过去的 M 分钟内，每隔 N 分钟去查询一次监控指标；

percent\_threshold: 配置阈值百分比，在监控时间内，报警次数占比大于该值即触发报警，发送报警邮件；

name: 配置报警时的邮件标题。

综合各参数的设置，在过去的 interval 时间段内，每隔 step 分钟去查询一次，即查询总次数为 interval/step（向下取整），设为 P；在监控的总次数内，监测值达到阈值表达式时的次数为 Q，则报警占比为 Q/P；如 Q/P\*100 大于阈值百分比 percent\_threshold，则触发报警，如 Q/P\*100 小于阈值百分比 percent\_threshold，则不触发报警。

```
cordon_dry_run: {{cnf['repair-manager']['ecc_rule']['cordon_dry_run']}}
prometheus:
  ip: {{cnf['repair-manager']['prometheus-ip']}}
  port: {{cnf['repair-manager']['prometheus-port']}}
metrics:
  - query_expression: 'nvidiasmi_ecc_error_count{type="volatile_double"}>0'
    step: 1m
    interval: 10 # minutes
    percent_threshold: 90 # percentage of data points with ecc error to be considered a bad node
    name: nvidiasmi_ecc_error_count
  - query_expression: '(100 - (avg by (instance)(irate(node_cpu_seconds_total{mode="idle"}[300s])) * 100))>90'
    step: 1m
    interval: 10 # minutes
    percent_threshold: 90 # percentage of data points with ecc error to be considered a bad node
    name: "cpu utilization more than 80%"
  - query_expression: '(node_memory_MemTotal_bytes - node_memory_MemFree_bytes - node_memory_Buffers_bytes - node_memory_Cached_bytes)/
    node_memory_MemTotal_bytes > 0.8'
    step: 1m
    interval: 10 # minutes
    percent_threshold: 90 # percentage of data points with ecc error to be considered a bad node
    name: "memory utilization more than 80%"
  - query_expression: 'avg (node_filesystem_free_bytes{fstype="nfs4"} / node_filesystem_size_bytes * 100) by (device, mountpoint,instance) <= 20'
    step: 1m
    interval: 10 # minutes
    percent_threshold: 90 # percentage of data points with ecc error to be considered a bad node
    name: "filesystem left space less than 20%"
```

图 49 ecc-config.yaml

### 3.8.3 rule-alerts.yaml

alerts\_enabled: 配置是否发送邮件，设置为 True，则发送报警邮件，设置为 False，则不发送报警邮件；

reminders\_enabled: 配置是否重复发送报警邮件，设置为 True，则会重复发送报警邮件，设置为 False，则只有第一次报警时才发送报警邮件；

reminder\_wait\_time: 配置重新发送报警邮件的时间间隔，如 reminders\_enabled 设置为 True，两次报警时间间隔大于 reminder\_wait\_time 则会再次发送报警邮件，如时间间隔小于 reminder\_wait\_time 则不会重复发送报警邮件。

```
ecc_rule:
  alerts_enabled: True # send email alerts
  reminders_enabled: True
  reminder_wait_time: 86400 # time to wait before resending email alert (seconds)
  alert_job_owners: False # alert all impacted job owners

cache-ttl: 600 # time until cache expires (seconds)
```

图 50 rule-alerts.yaml

### 3.8.4 rule-config.yaml

rule\_wait\_time: 设置查询间隔，单位为秒，该参数设置为 N，即与上次查询各指标的时间间隔为 N 秒。

```
---
rules:
  ecc_rule:
    module_name : rules.ecc_rule
    class_name : ECCRule

rule_wait_time: 10 # time to sleep between rule execution (seconds)

cluster_name: {{cnf['repair-manager']['cluster_name']}}
```

图 51 rule-config.yaml