```
provider "aws" {
  region = "us-west-2" # Choose an appropriate region
}

resource "aws_security_group" "allow_http" {
  name        = "allow_http"
  description = "Allow HTTP inbound traffic"

  ingress {
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}

resource "aws_launch_template" "gpu_instance" {
  name_prefix   = "gpu-instance-"
  image_id      = "ami-123456" # Specify an appropriate AMI for your GPU instance
```

```
  instance_type = "p3.2xlarge" # This is an example GPU instance type. Adjust based on your needs.

  network_interfaces {
    associate_public_ip_address = true
    security_groups             = [aws_security_group.allow_http.id]
  }

  user_data = <<-EOF
              #!/bin/bash
              echo "Your setup script goes here. This could install Docker, your application, etc."
              EOF
}

resource "aws_autoscaling_group" "gpu_asg" {
  launch_template {
    id      = aws_launch_template.gpu_instance.id
    version = "$Latest"
  }

  min_size         = 1
  max_size         = 3
  desired_capacity = 2
  vpc_zone_identifier = ["subnet-12345"] # Specify your subnet IDs

  tag {
```

```
    key                 = "Name"
    value               = "GPUInstance"
    propagate_at_launch = true
  }
}

resource "aws_elb" "api_load_balancer" {
  name               = "api-lb"
  availability_zones = ["us-west-2a", "us-west-2b"] # Adjust based on your VPC and subnet setup

  listener {
    instance_port     = 80
    instance_protocol = "HTTP"
    lb_port           = 80
    lb_protocol       = "HTTP"
  }

  health_check {
    target              = "HTTP:80/"
    interval            = 30
    timeout             = 5
    healthy_threshold   = 2
    unhealthy_threshold = 2
  }

  instances                   = [aws_autoscaling_group.gpu_asg.*.id]
```

```
  cross_zone_load_balancing   = true
  idle_timeout                = 400
  connection_draining         = true
  connection_draining_timeout = 400
}
```