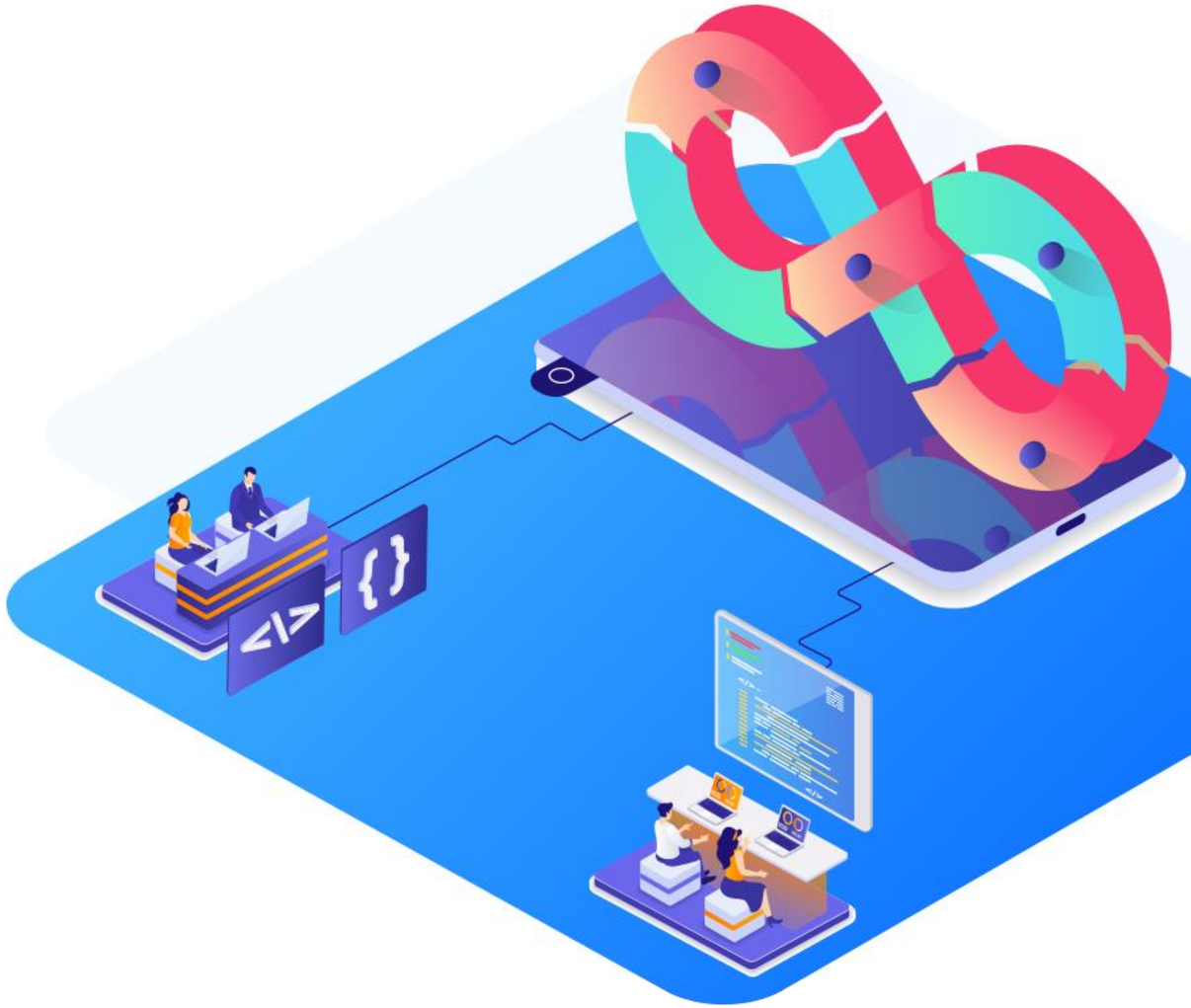
Container Orchestration Using Kubernetes



Course-End Project

Deploy an Application Using the Kubernetes Dashboard

Objective

To deploy a multi-tier MySQL application using Kubernetes with specific configurations for user roles, storage, service verification, namespace restrictions, quota limits, and data management

Problem Statement and Motivation

Real-time scenario:

Karen is a DevOps engineer at a tech startup. Her team has developed a new application using MySQL. Now, it is her task to deploy that application.

The company plans to utilize Kubernetes for its robust container orchestration capabilities.

Karen must create a Kubernetes dashboard with specific configurations, user roles, storage, service verification, and data management.

Industry Relevance

The following tools used in this project serve specific purposes within the industry:

1. **kubeadm:** A utility that offers **kubeadm init** and **kubeadm** join as efficient ways to bootstrap Kubernetes clusters. It focuses on bootstrapping rather than machine provisioning.

2. **kubectl:** A command-line interface for Kubernetes that allows execution of commands against Kubernetes clusters. It can be used for deploying applications, managing cluster resources, and viewing logs.

3. **kubelet:** An essential node agent present on every node in a Kubernetes cluster. It ensures that the containers described in the provided PodSpecs are running and healthy.

4. **Docker:** Docker is a tool designed to facilitate developers in building, sharing, and running applications in containers. It takes care of the setup, allowing developers to concentrate on the code.

Tasks

The following tasks outline the process of deploying an application using Kubernetes:

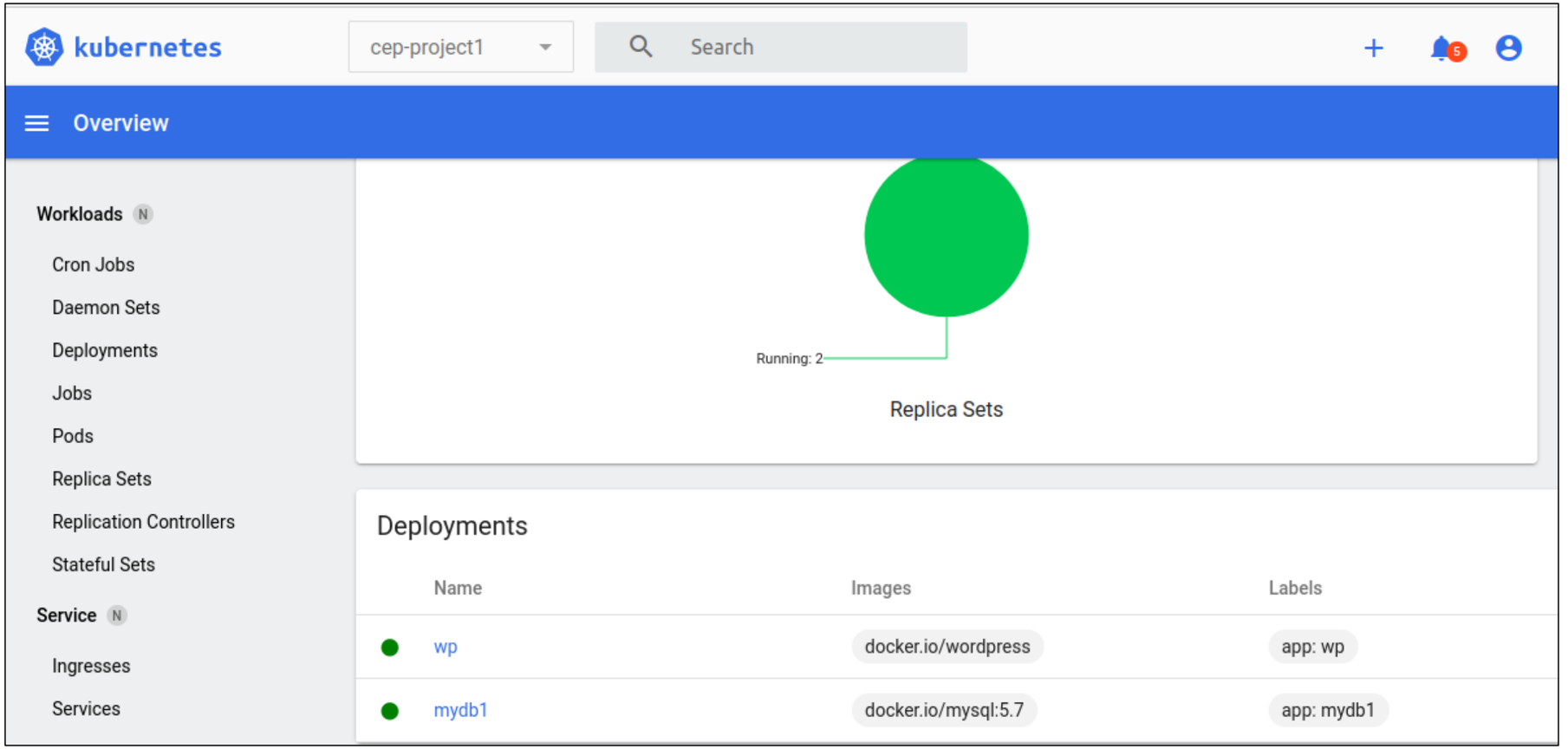
1. Get started with pods, services, and deployments
2. Create and verify the service
3. Create a token and work on a dashboard
4. Configure the NFS-server for MySQL and WordPress deployment
5. Set up the NFS client side
6. Create and verify the PV
7. Create a secret for MySQL deployments secret data
8. Create a configmap for WordPress deployment to store non-sensitive information

Project References

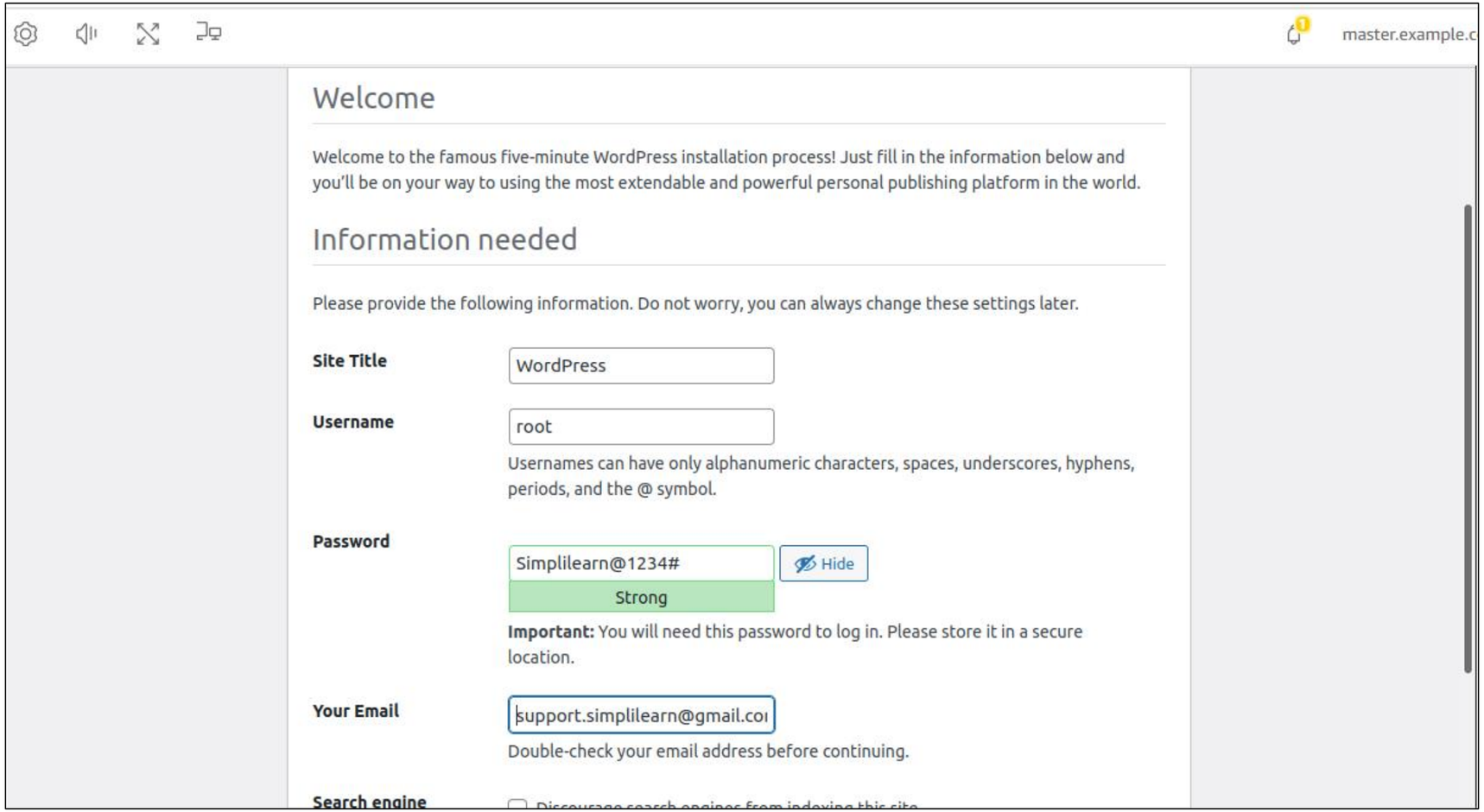
Task 1: Lesson 2
Task 2: Lesson 6
Task 3: Lesson 3
Task 4: Lesson 4
Task 5: Lesson 4
Task 6: Lesson 7
Task 7: Lesson 4
Task 8: Lesson 4

COURSE-END PROJECT

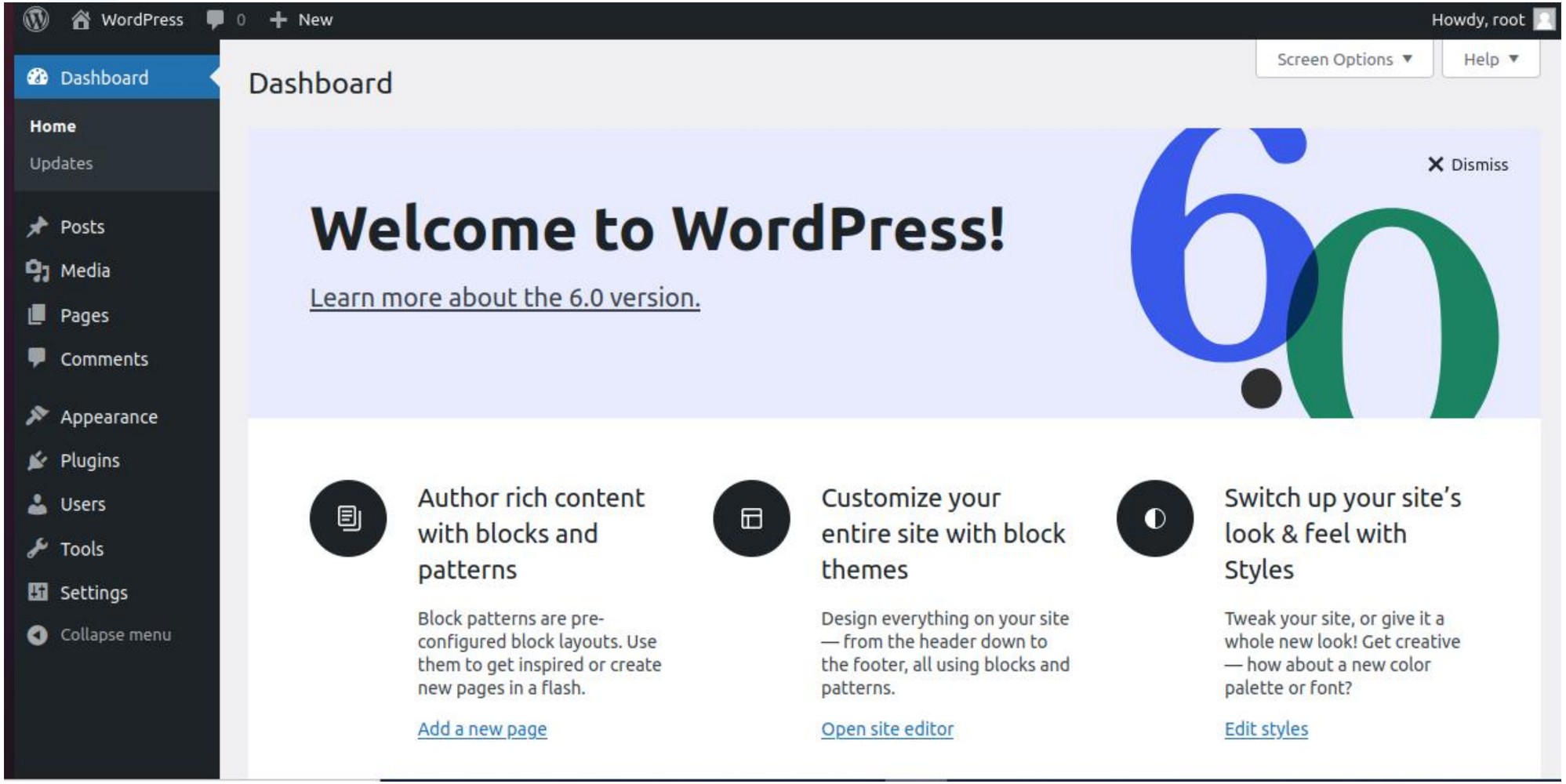
Output Screenshots



Output Screenshots



Output Screenshots



Thank you