|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Инструментального и прикладного программного обеспечения

**ПРАКТИЧЕСКАЯ РАБОТА №5**

по дисциплине «Технологии виртуализации клиент-серверных приложений»

**Студент группы** ИКБО-20-19 Анваржонов Ж. Т

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы** преподаватель Волков М.Ю.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва 2022

# **Постановка задачи**

* Установить minukube, собрать и запустить контейнер, сервис в одноузловом кластере kubernetes.

# **Выполнение работы**

Установим kubectl

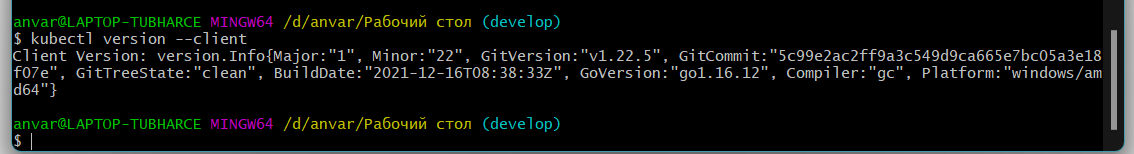


Рисунок – установка kubectl

Установка minikube

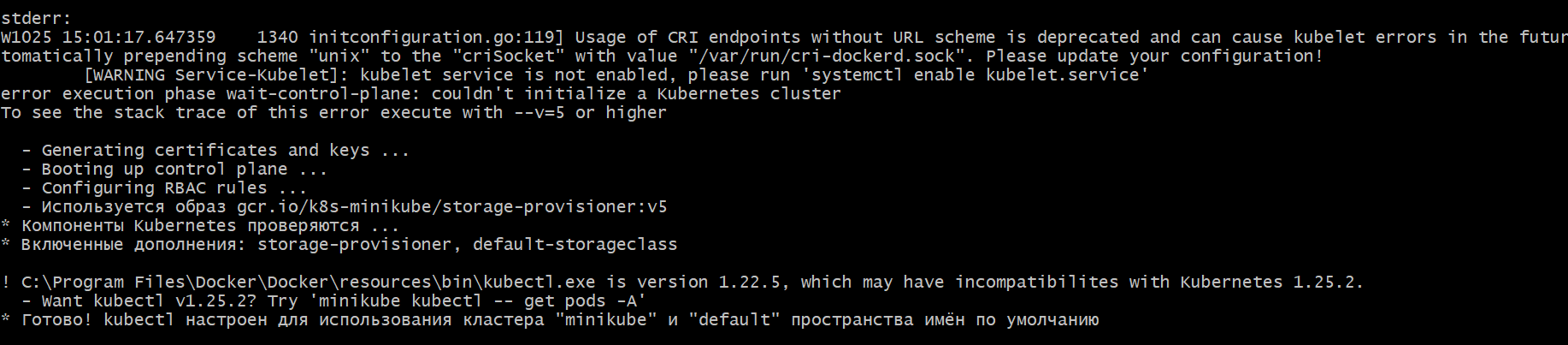


Рисунок – процесс установки minikube

Статус одногоузлового кластера kubernetes с помощью minikube команды

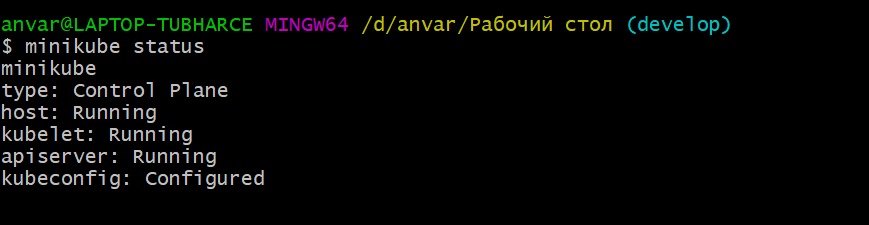


Рисунок – статус minikube

Соберем образ на основе node для запуска server.js

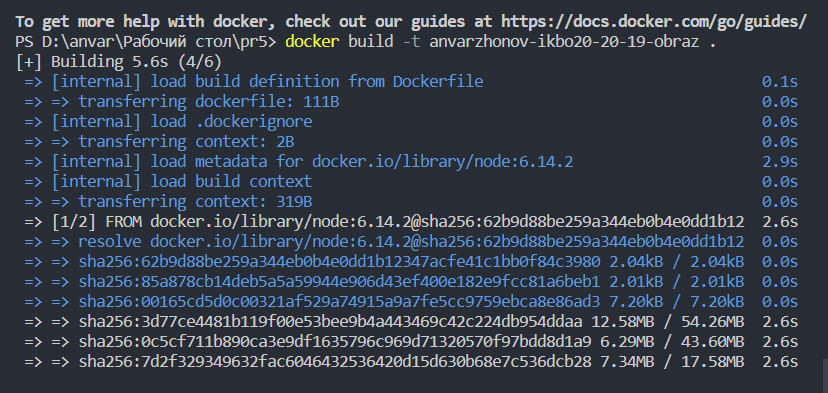


Рисунок – процесс сборки образа

Создадим конфигурацию для развертывания данного образа. Для этого создадим файл deployment.yml

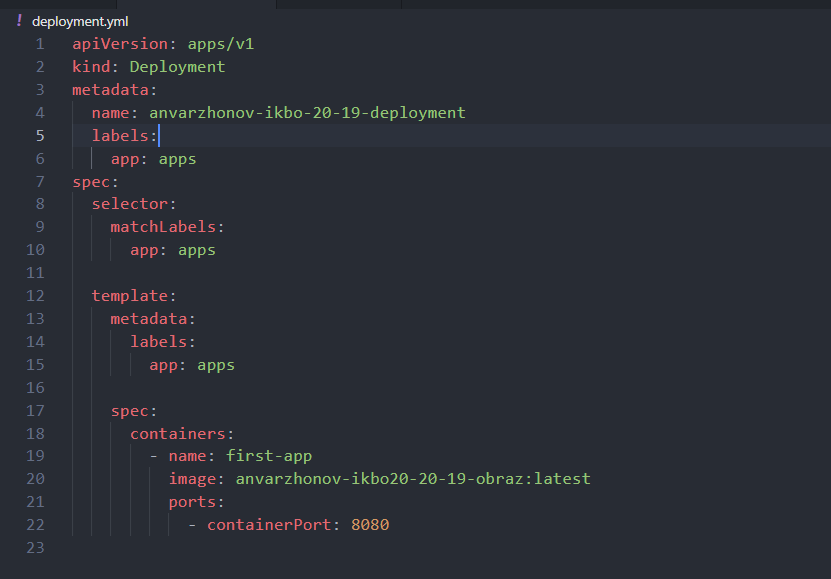


Рисунок – конфигурация deployment.yml

Применим данную конфигурацию и посмотрим созданные объекты



Рисунок – применение deployment.yml

Как видим возникает ошибка при пуле созданного на локальной машине образа. Это происходит, потому что в docker registry кластера minikube нет данного образа. Прокинем наш локальный docker в minukube:

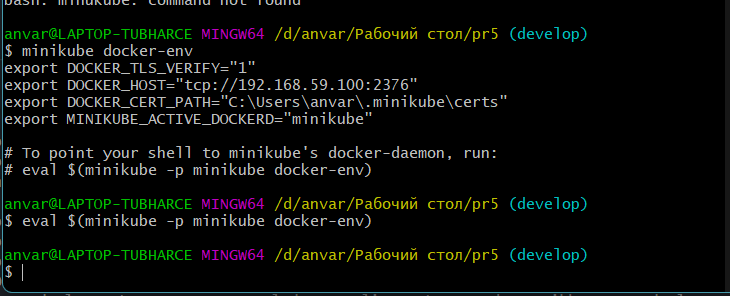


Рисунок – Правки minikube

Добавим правки в deployment.yml, чтобы образ не тянулся из общедоступного docker внутри minikube

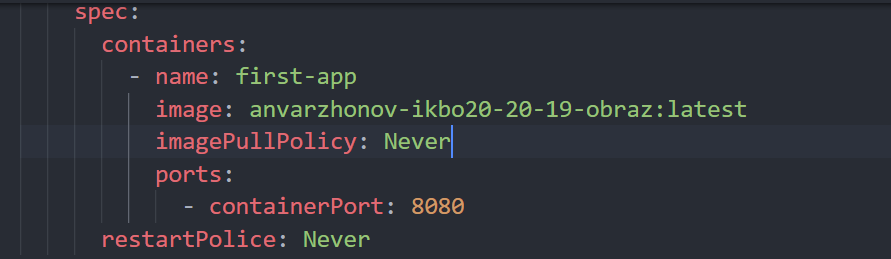


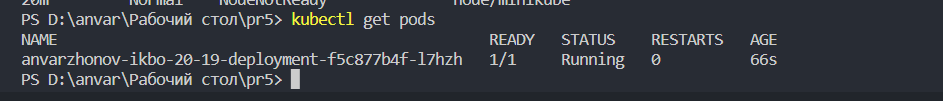
Рисунок – правки deployment.yml

Пересоберем наш образ заново, чтобы он появился в registry minikube.



Рисунок – пересборка образа

Покажем успешно запущенный pod, его events и логи



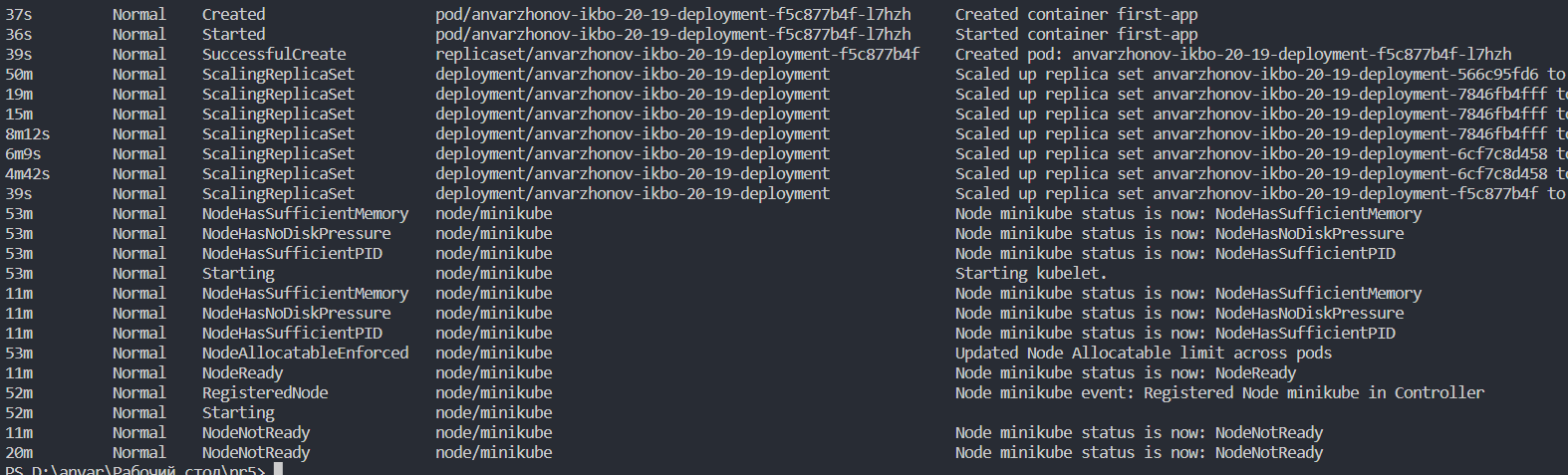


Рисунок – логи пода, евентов

Покажем kubectl конфигурацию

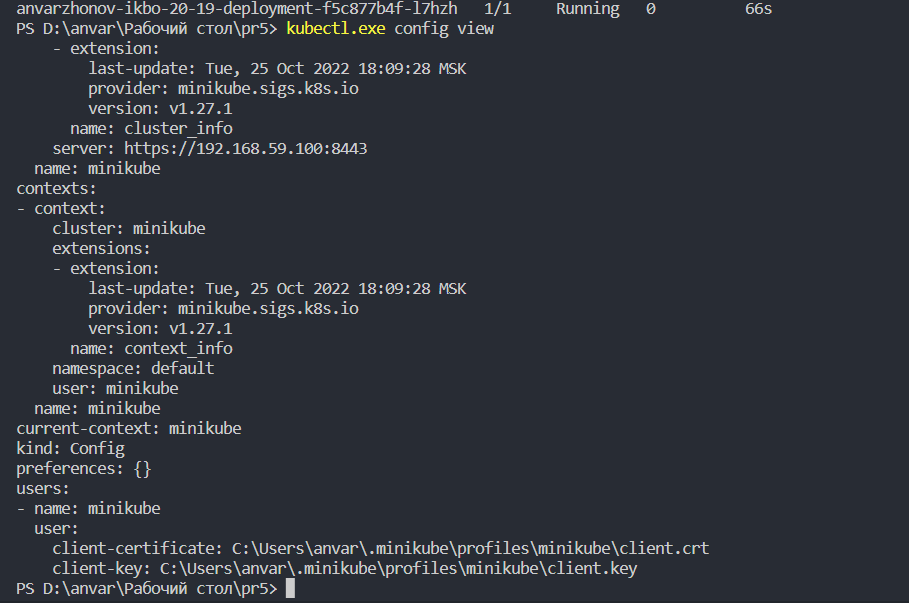


Рисунок – kubectl конфигурация

Cделаем под доступным для публичной сети Интернет c помощью команды kubectl expose:

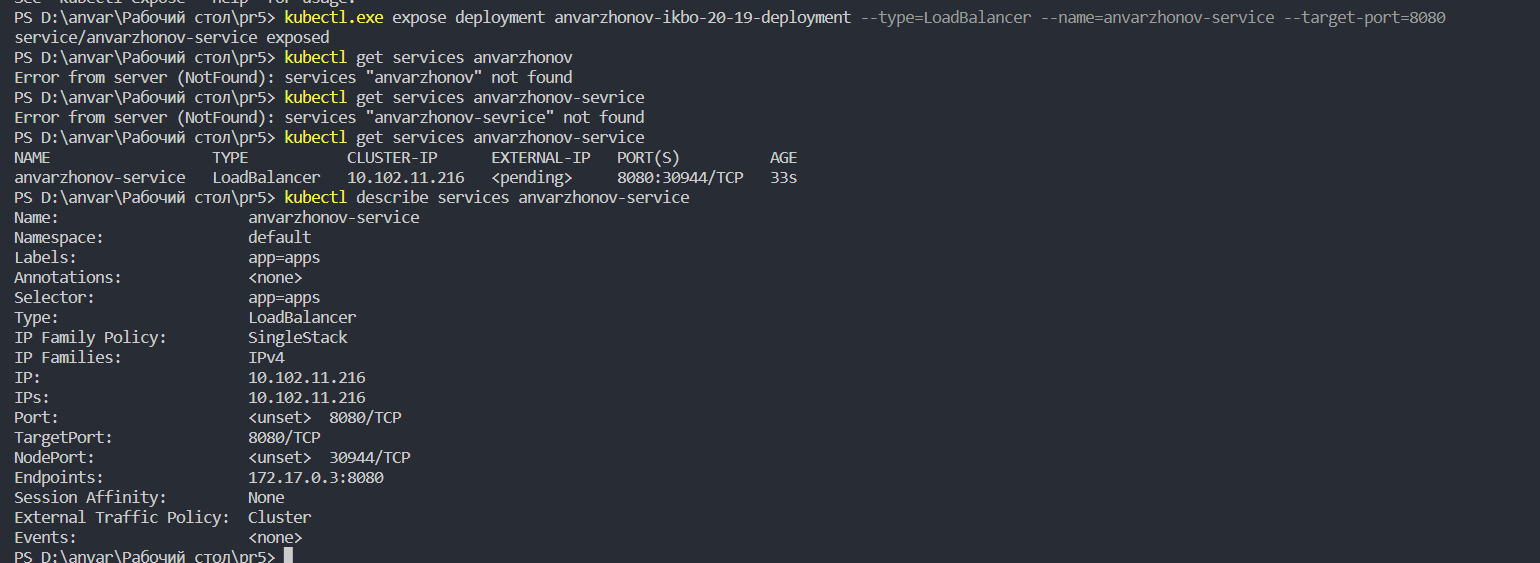
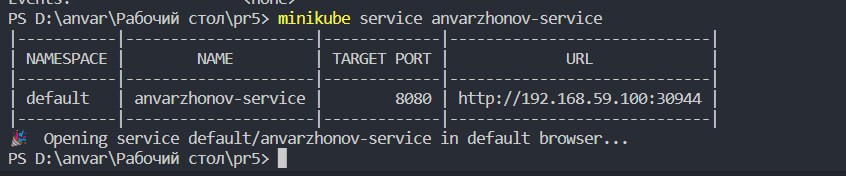


Рисунок –настройка сервиса

Запустим наш сервис:



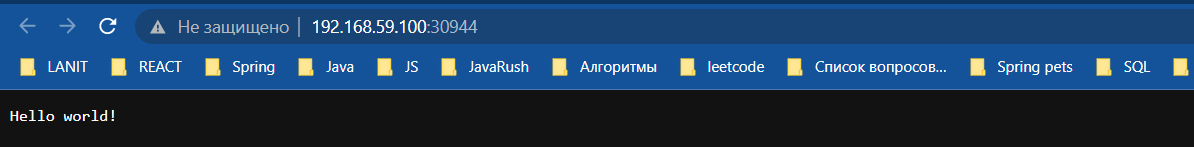
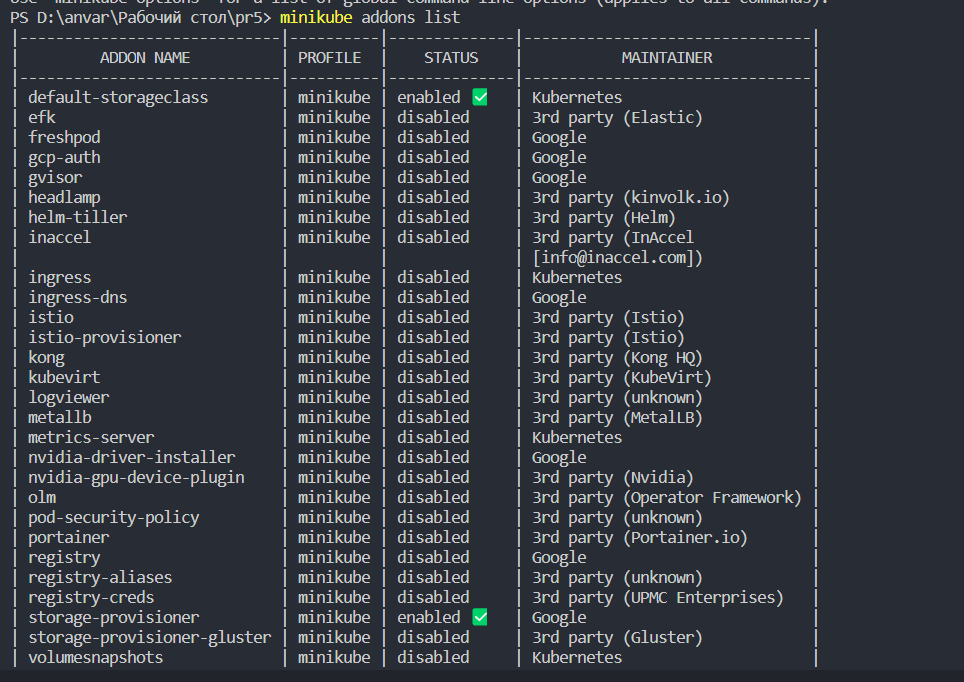


Рисунок – запуск сервиса

Покажем список доступных addons и включим ingress



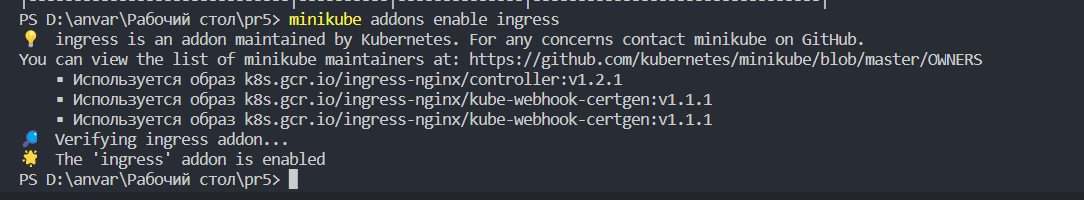


Рисунок – список addons

Покажем список системных pod и service:

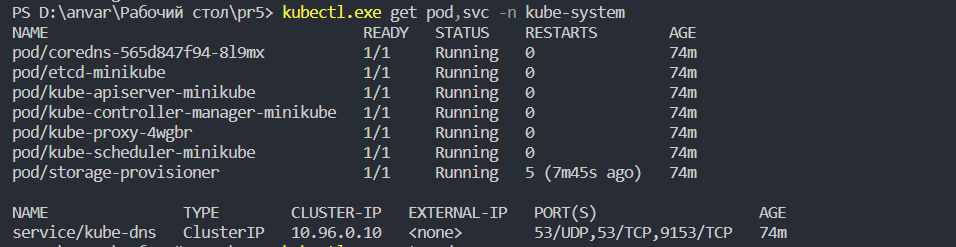
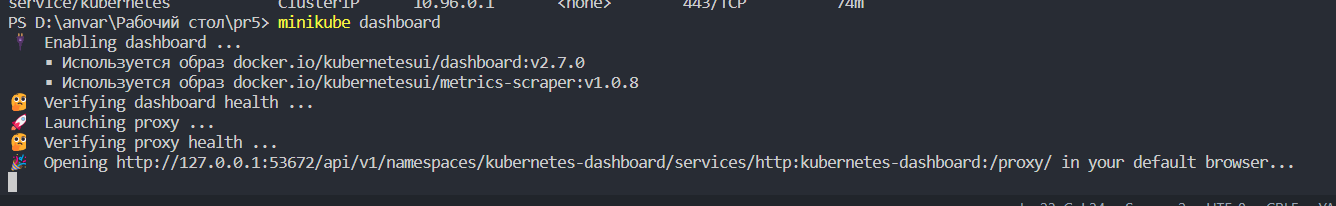
****

Рисунок - отображение системных подов и сервисов

Включим dashboard



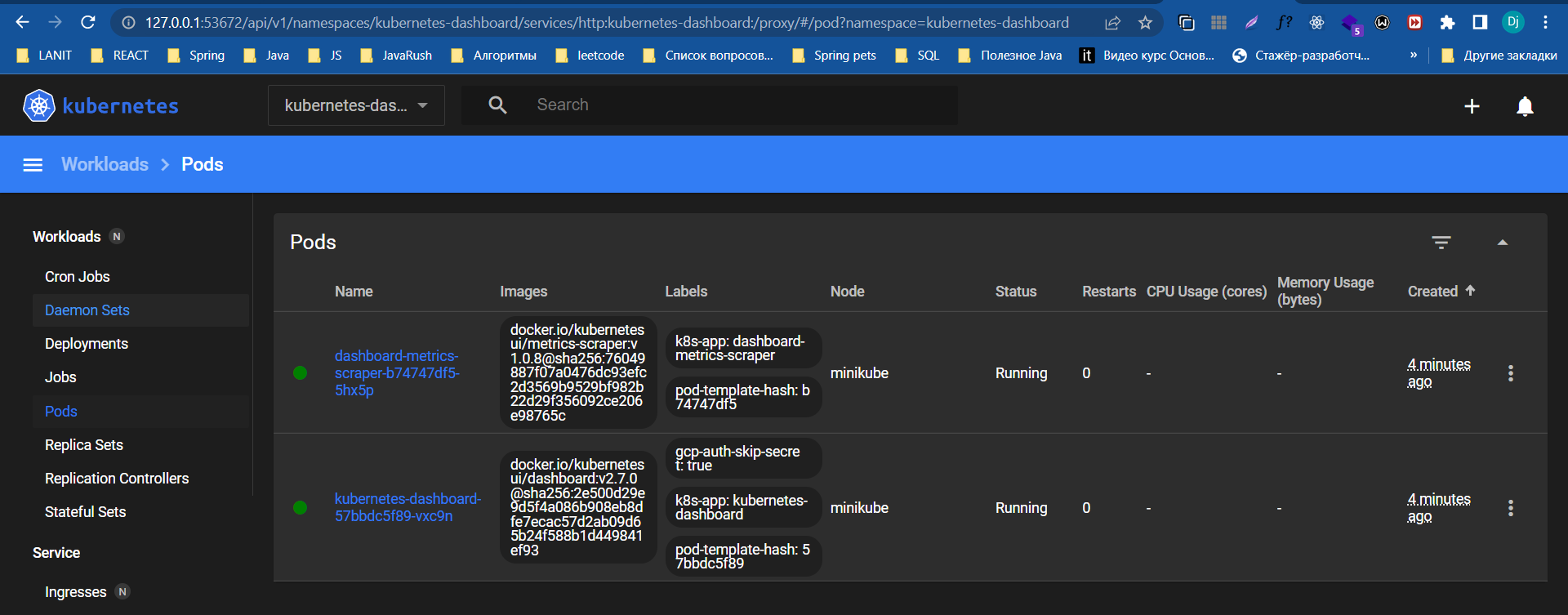


Рисунок – отображение дашборда

Остановим minikube:

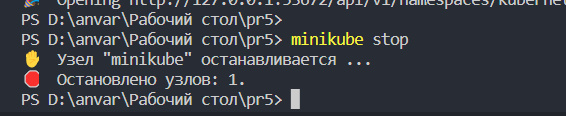


Рисунок – остановка миникуба

**Вывод**

В результате выполнения четвертой практической работы были получены навыки работы с Kubernetes и minikube , были созданы поды и сервисы для их работы.

**Ответы на вопросы к практической работе**

1. Назовите виды контроллеров в Kubernetes.

*Deployments* - контроллер, который управляет состоянием развертывания подов, которое описывается в манифесте, следит за удалением и созданием экземпляров подов. Управляет контроллерами ReplicaSet.

*ReplicaSet* - гарантирует, что определенное количество экземпляров подов всегда будет запущено в кластере.

*StatefulSets* - так же как и Deployments, управляет развертыванием и масштабированием набора подов, но сохраняет набор идентификаторов и состояние для каждого пода.

*DaemonSet* - гарантирует, что на каждом узле кластера будет присутствовать экземпляр пода.

*Jobs* - создает определенное количество подов и смотрит, пока они успешно не завершат работу. Если под завершился с ошибкой, повторяет создание, которое мы описали определенное количество раз. Если под успешно отработал, записывает это в свой журнал.

*CronJob* - запускает контроллеры Jobs по определенному расписанию.

2. Как называется командная строка в Kubernetes?

Kubectl — это инструмент командной строки для управления кластерами Kubernetes. kubectl ищет файл config в директории $HOME/.kube. Вы можете указать другие файлы kubeconfig, установив переменную окружения KUBECONFIG или флаг --kubeconfig.

3. Что такое под?

Pods или поды — это абстрактный объект в кластере K8S, который состоит из одного или нескольких контейнеров с общим хранилищем и сетевыми ресурсами, а также спецификации для запуска контейнеров.

Это главный объект в кластере, в нем прописаны, какие контейнеры должны быть запущены, количество экземпляров или реплик, политика перезапуска, лимиты, подключаемые ресурсы, узел кластера для размещения.

kube-scheduler планирует размещение пода на узлах кластера

kubelet на рабочем узле кластера запускает под

4. Назовите 2 типа ресурсов, из которых состоит кластер Kubernetes.

Сам кластер K8S состоит из, барабанная дробь, рабочих узлов. В узлах или нодах (Nodes, Worker nodes), помимо контейнеров компонентов самого кластера, размещаются контейнеры наших проектов и сервисов.

Worker nodes состоит из компонентов:

kubelet - сервис или агент, который контролирует запуск компонентов (контейнеров) кластера

kube-proxy - конфигурирует правила сети на узлах

Плоскость управления (Master nodes) управляет рабочими узлами и подами в кластере. Там располагаются компоненты, которые управляют узлами кластера и предоставляют доступ к API.

Control plane состоит из компонентов:

kube-apiserver - предоставляет API кубера

etcd - распределенное key-value хранилище для всех данных кластера. Необязательно располагается внутри мастера, может стоять как отдельный кластер

kube-scheduler - планирует размещение подов на узлах кластера

kube-controller-manager - запускает контроллер

kubelet - сервис или агент, который контролирует запуск основных компонентов (контейнеров) кластер

5. Чем Kubernetes отличается от Docker Swarm?

Kubernetes уже включает множество инструментов оркестровки, например, автомасштабирование.