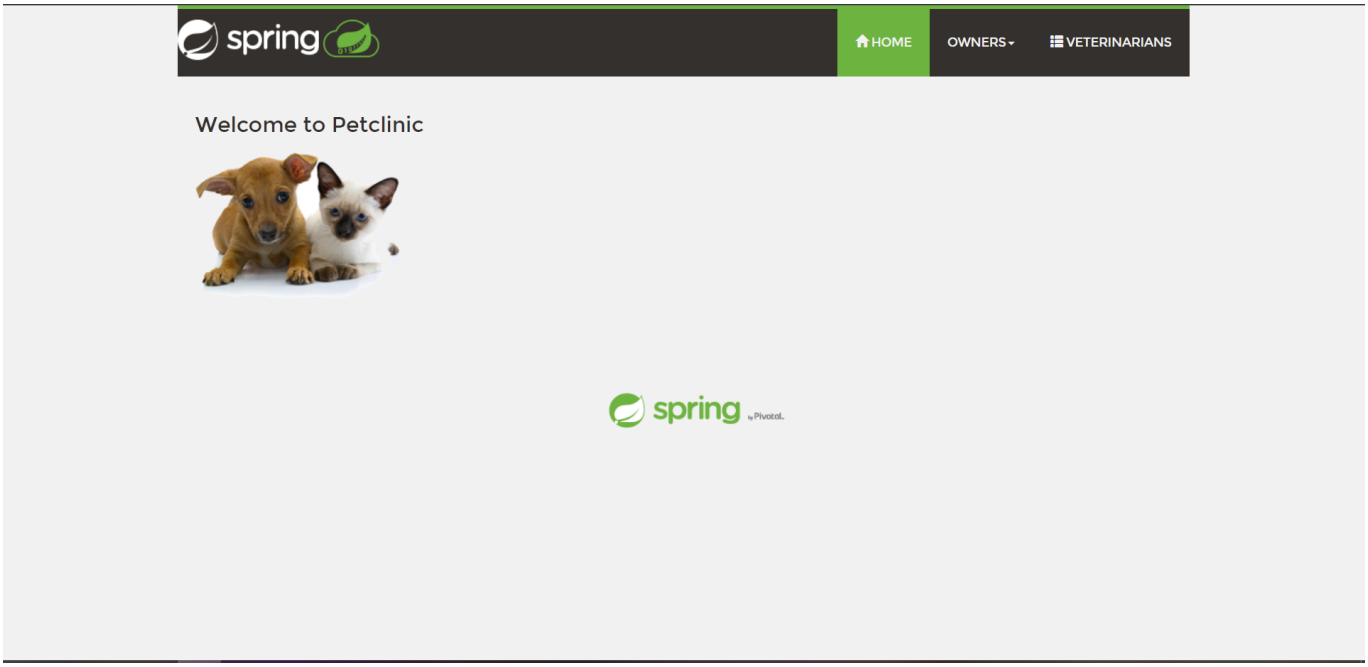
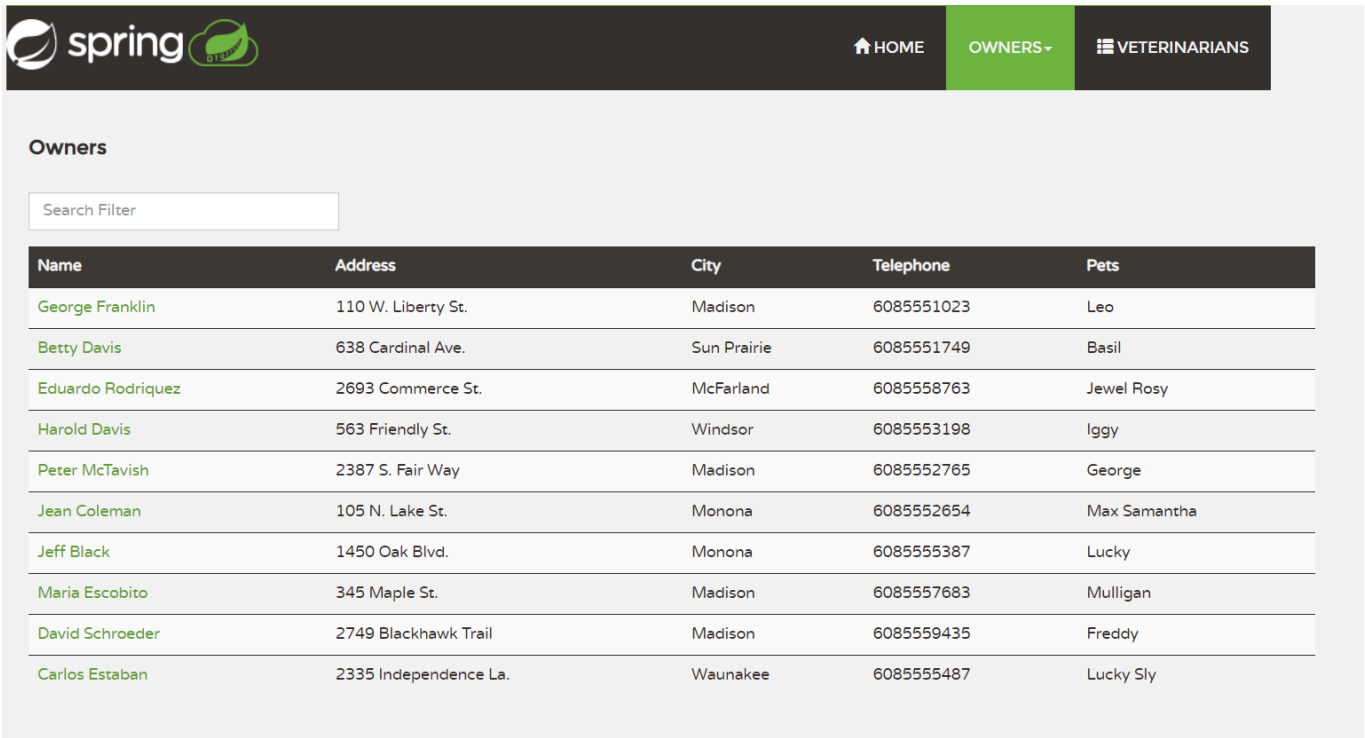
lab 3 Damian Pałyska Tomasz Wężowicz

podstawowa konfiguracja:

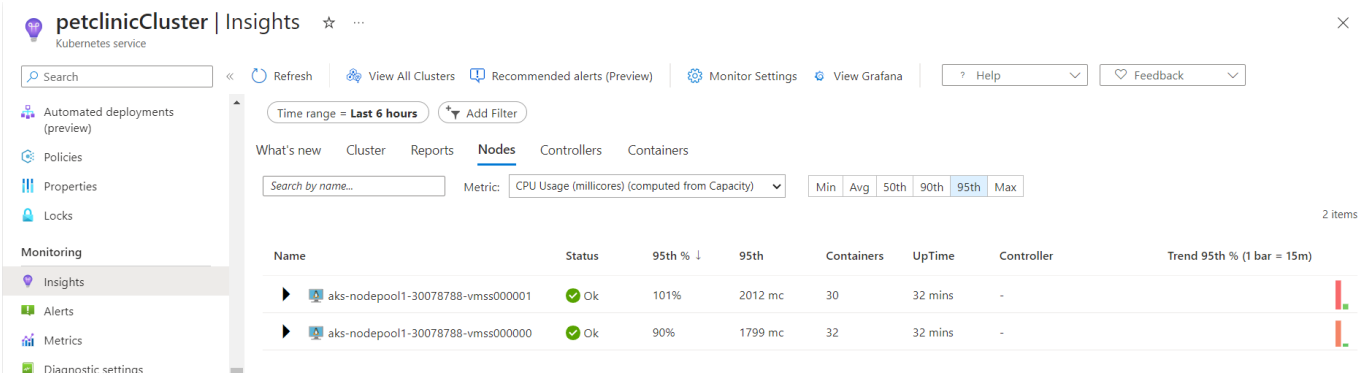
Tworzymy kluster kubernetesowy za pomocą pliku scirpt.sh następnie wykonujemy deploy ze skryptu deploy_app.sh

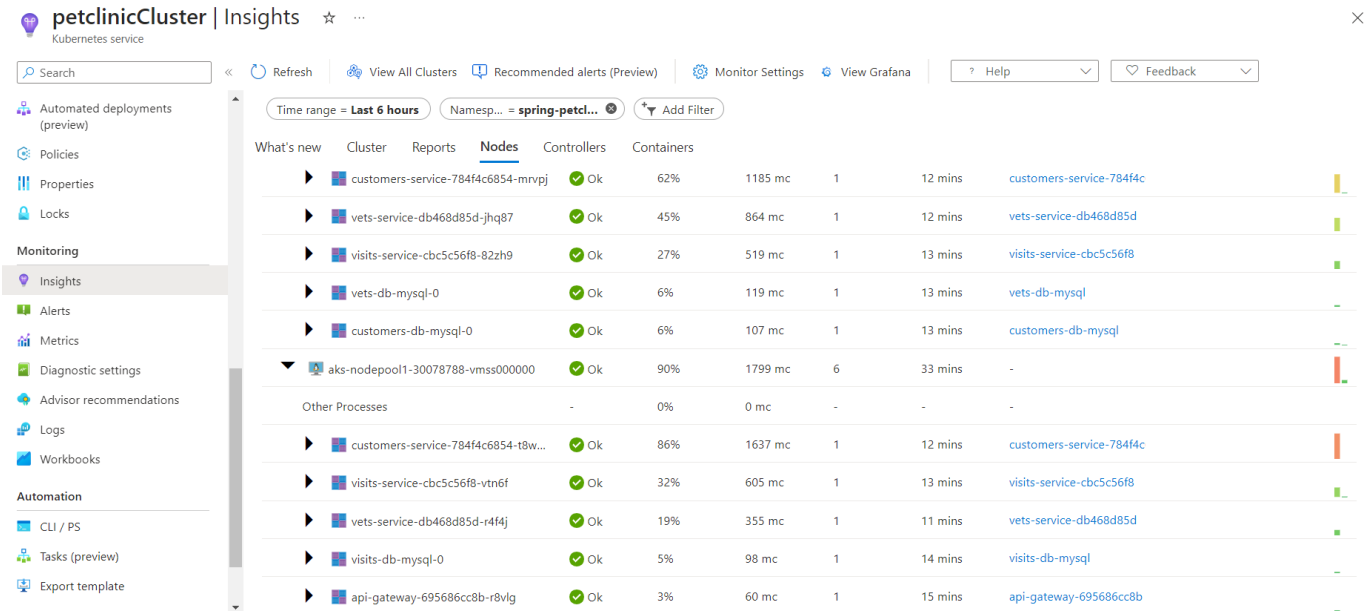


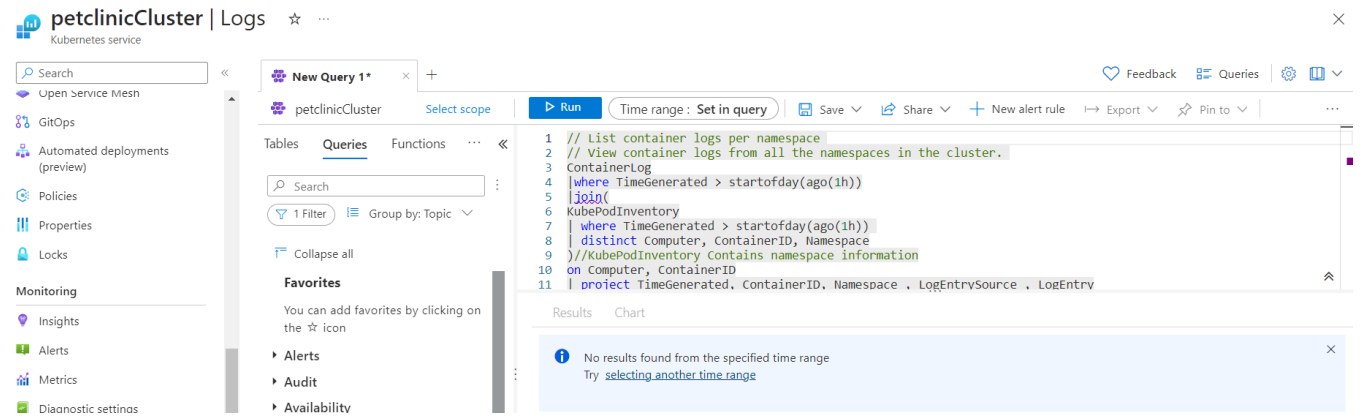


Name	Address	City	Telephone	Pets
George Franklin	110 W. Liberty St.	Madison	6085551023	Leo
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085551749	Basil
Eduardo Rodriquez	2693 Commerce St.	McFarland	6085558763	Jewel Rosy
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George
Jean Coleman	105 N. Lake St.	Monona	6085552654	Max Samantha
Jeff Black	1450 Oak Blvd.	Monona	6085555387	Lucky
Maria Escobito	345 Maple St.	Madison	6085557683	Mulligan
David Schroeder	2749 Blackhawk Trail	Madison	6085559435	Freddy
Carlos Estaban	2335 Independence La.	Waunakee	6085555487	Lucky Sly

azure insights


petclinicCluster | Insights
Kubernetes service
Search
Refresh
View All Clusters
Recommended alerts (Preview)
Monitor Settings
View Grafana
Help
Feedback
Automated deployments (preview)
Policies
Properties
Locks
Monitoring
Insights
Alerts
Metrics
Diagnostic settings
Time range = Last 6 hours
Add Filter
What's new
Cluster
Reports
Nodes
Controllers
Containers
Search by name...
Metric:
CPU Usage (millicores) (computed from Capacity)
Min
Avg
50th
90th
95th
Max
2 items
Name
Status
95th %
95th
Containers
UpTime
Controller
Trend 95th % (1 bar = 15m)
aks-nodepool1-30078788-vmss000001
Ok
101%
2012 mc
30
32 mins
aks-nodepool1-30078788-vmss000000
Ok
90%
1799 mc
32
32 mins


petclinicCluster | Insights
Kubernetes service
Search
Refresh
View All Clusters
Recommended alerts (Preview)
Monitor Settings
View Grafana
Help
Feedback
Automated deployments (preview)
Policies
Properties
Locks
Monitoring
Insights
Alerts
Metrics
Diagnostic settings
Advisor recommendations
Logs
Workbooks
Automation
CLI / PS
Tasks (preview)
Export template
Time range = Last 6 hours
Namesp... = spring-petcl...
Add Filter
What's new
Cluster
Reports
Nodes
Controllers
Containers
customers-service-784f4c6854-mrvpj
Ok
62%
1185 mc
1
12 mins
customers-service-784f4c
vets-service-db468d85d-jhq87
Ok
45%
864 mc
1
12 mins
vets-service-db468d85d
visits-service-cbc5c56f8-82zh9
Ok
27%
519 mc
1
13 mins
visits-service-cbc5c56f8
vets-db-mysql-0
Ok
6%
119 mc
1
13 mins
vets-db-mysql
customers-db-mysql-0
Ok
6%
107 mc
1
13 mins
customers-db-mysql
aks-nodepool1-30078788-vmss000000
Ok
90%
1799 mc
6
33 mins
Other Processes
0%
0 mc
customers-service-784f4c6854-t8w...
Ok
86%
1637 mc
1
12 mins
customers-service-784f4c
visits-service-cbc5c56f8-vtn6f
Ok
32%
605 mc
1
13 mins
visits-service-cbc5c56f8
vets-service-db468d85d-r4f4j
Ok
19%
355 mc
1
11 mins
vets-service-db468d85d
visits-db-mysql-0
Ok
5%
98 mc
1
14 mins
visits-db-mysql
api-gateway-695686cc8b-r8vlg
Ok
3%
60 mc
1
15 mins
api-gateway-695686cc8b


petclinicCluster | Logs
Kubernetes service
Search
Open Service Mesh
GitOps
Automated deployments (preview)
Policies
Properties
Locks
Monitoring
Insights
Alerts
Metrics
Diagnostic settings
New Query 1*
petclinicCluster
Select scope
Run
Time range : Set in query
Save
Share
New alert rule
Export
Pin to
Feedback
Queries
Tables
Queries
Functions
Search
1 Filter
Group by: Topic
Collapse all
Favorites
You can add favorites by clicking on the ☆ icon
Alerts
Audit
Availability
// List container logs per namespace
// View container logs from all the namespaces in the cluster.
ContainerLog
|where TimeGenerated > startofday(ago(1h))
|join(
KubePodInventory
| where TimeGenerated > startofday(ago(1h))
| distinct Computer, ContainerID, Namespace
)//KubePodInventory Contains namespace information
on Computer, ContainerID
| project TimeGenerated, ContainerID, Namespace , LogEntrySource , LogEntry
Results
Chart
No results found from the specified time range
Try selecting another time range

terraform

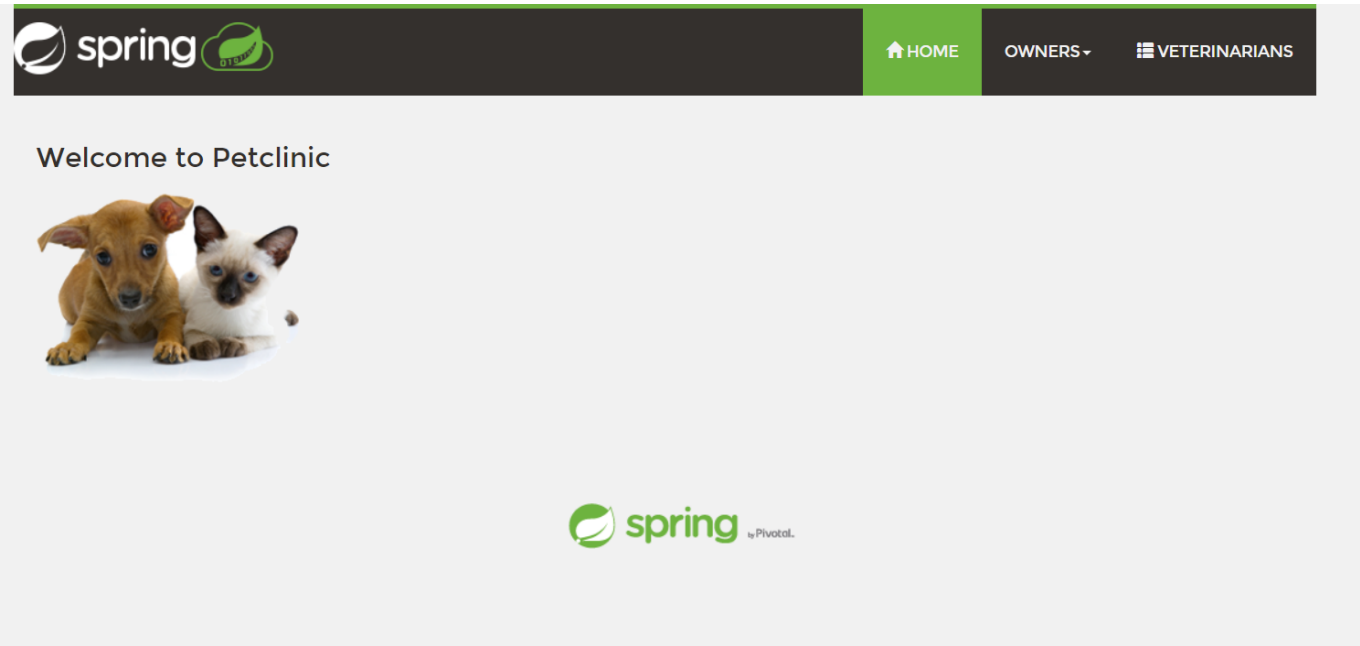
```
azureuser [ ~ ]$ terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # azurerm_application_insights.insights will be created
  + resource "azurerm_application_insights" "insights" {
      + app_id                              = (known after apply)
      + application_type                    = "web"
      + connection_string                   = (sensitive value)
      + daily_data_cap_in_gb                = (known after apply)
      + daily_data_cap_notifications_disabled = (known after apply)
      + disable_ip_masking                  = false
      + force_customer_storage_for_profiler = false
      + id                                  = (known after apply)
      + instrumentation_key                 = (sensitive value)
      + internet_ingestion_enabled          = true
      + internet_query_enabled              = true
      + local_authentication_disabled       = false
      + location                            = "westeurope"
      + name                                = "petclinic-appinsights"
      + resource_group_name                 = "wus_3"
      + retention_in_days                   = 90
      + sampling_percentage                 = 100
      + workspace_id                        = (known after apply)
    }

  # azurerm_kubernetes_cluster.aks will be created
  + resource "azurerm_kubernetes_cluster" "aks" {
      + dns_prefix                        = "petclinicCluster"
      + fqdn                              = (known after apply)
```

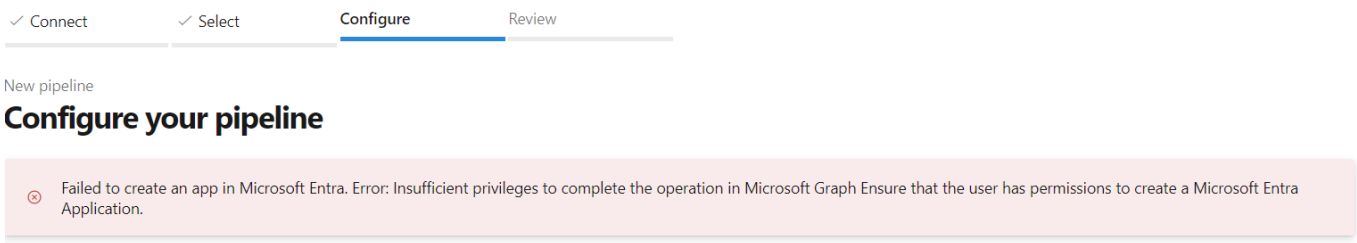


Tworzymy kluster kubernetesowy z pliku main.tf komendami: terraform init, terraform validate, terraform apply następnie wykonujemy deploy ze skryptu deploy_app.sh

pipelines

Pomimo dodania nowego rejestru komendą:

az acr create --resource-group wus_3 --name mydpcontainerregistry --sku Basic



nie udało nam się stworzyć pipeline, przy próbie pokazuje się taki błąd.