

# Kubernetes Alapú Ingatlan Marketplace Infrastruktúra

## Feladatkiírás

### Feladat leírása

Egy egyszerűsített Kubernetes-alapú infrastruktúra létrehozása egy ingatlan marketplace platformhoz Terraform segítségével, valamint egy alapvető CI/CD pipeline kialakítása. A cél egy gyorsan telepíthető és jól skálázható Kubernetes-megoldás kialakítása.

A feladat megoldására – becslésünk szerint – körülbelül 3-5 órát érdemes rászánni. Nem az a legfontosabb, hogy a feladat teljes, minden részletre kiterjedő, hiba nélkül működő megoldás legyen. Sokkal inkább arra vagyunk kíváncsiak, miként gondolkodsz a problémáról, milyen kreatív ötleteid vannak, és hogyan közelíted meg a megvalósítást, milyen szempontok alapján hozol döntéseket, és hogyan biztosítanád a rendszer továbbfejleszthetőségét, biztonságát és skálázhatóságát. Ha bizonyos elemeket nem tudsz teljes részletességgel implementálni, kérjük, vázold fel a koncepciódat, írd (vagy rajzold) le az elképzeléseidet, azaz, hogy hogyan oldanád meg az adott részt.

A marketplace a következő komponensekből áll:

- Azure SQL
- .NET Backend
- React Frontend
- Nginx, mint gateway

### Feladatok

- **Terraform infrastruktúra kialakítása:**
  - Készíts egy Terraform modult, amely egy Kubernetes klasztert hoz létre egy cloud providerben (pl. Azure AKS vagy AWS EKS).
  - Definiálj egy egyszerű, egy-node-os cluster setupot, amely később bővíthető.
  - Implementáld a Kubernetes konfigurációt Terraform segítségével (például kubernetes provider használatával).
- **Alap Kubernetes Deployment és Load Balancing:**
  - Hozz létre egy egyszerű Kubernetes Deployment-et a komponensekből.
  - Konfiguráld egy Service-t (LoadBalancer vagy NodePort típussal) az alkalmazás elérhetőségéhez.
- **CI/CD pipeline beállítása:**

- Írj egy egyszerű Azure Devops vagy GitLab CI/CD pipeline-t, amely:
  - Buildeli és pusholja a Docker image-et egy container registry-be (pl. Docker Hub, Azure Container Registry, AWS ECR).
  - Deployolja az alkalmazást a Kubernetes klaszterbe a kubectl apply vagy Helm segítségével.
- **Terraform state és modularitás:**
  - Használd távoli state kezelést (pl. Azure Storage).
  - Strukturáld úgy a Terraform kódot, hogy az könnyen bővíthető legyen (moduláris elrendezés).

## Elvárások

- **Terraform kód:** Egy egyszerű Kubernetes klaszter létrehozása Terraformmal.
- **Kubernetes konfiguráció:** Kubernetes deployment.
- **CI/CD pipeline:** Egy pipeline, amely buildeli és deployolja az alkalmazást.
- **Dokumentáció:** Egy rövid README fájl, amely magyarázza a megoldást és a futtatás lépéseit.

## Opcionális feladatok

- **Helm Chart vagy Kustomize:** Ezek helyett elég egy YAML manifest.
- **Autoscaling, Service Mesh, Advanced Security:** Csak alap RBAC és Secret kezelés.
- **Komplex monitoring (Prometheus, Grafana):** Egy egyszerű kubectl logs vagy Kubernetes Events használata elegendő.

## Bírálati szempontok

1. **Terraform kód helyessége és modularitása** (terraform state, provider helyes használata).
2. **Kubernetes deployment és networking megoldás** (helyesen működő Service és Deployment).
3. **CI/CD pipeline működőképessége és hatékonysága.**
4. **A megoldás dokumentáltsága és könnyen követhetősége.**