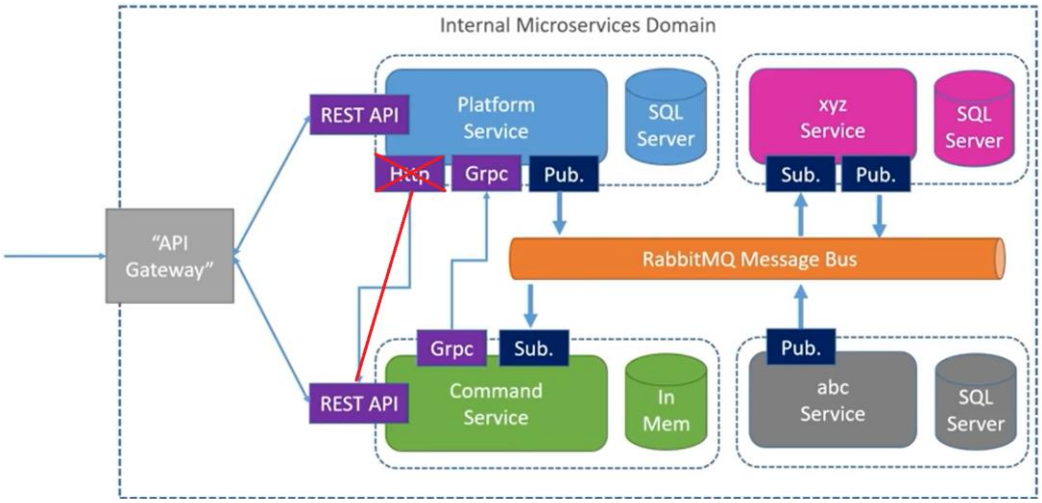
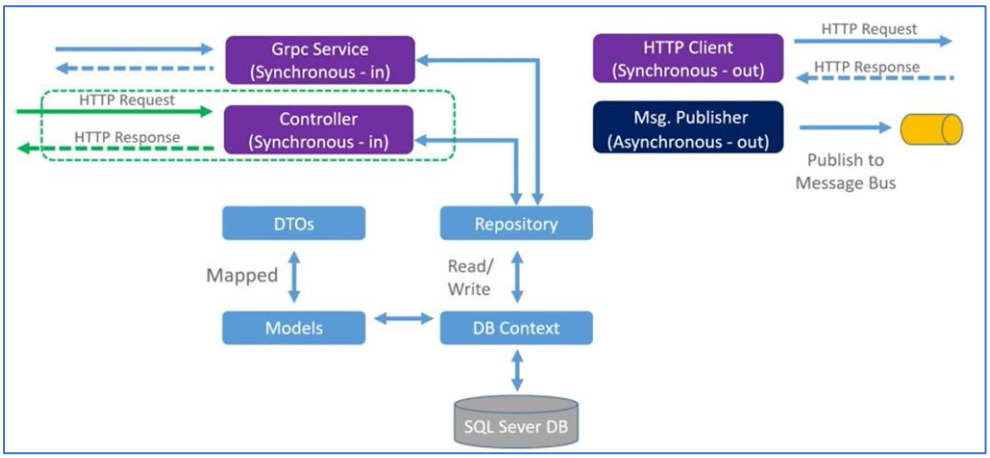
Microservices Best Practices Project

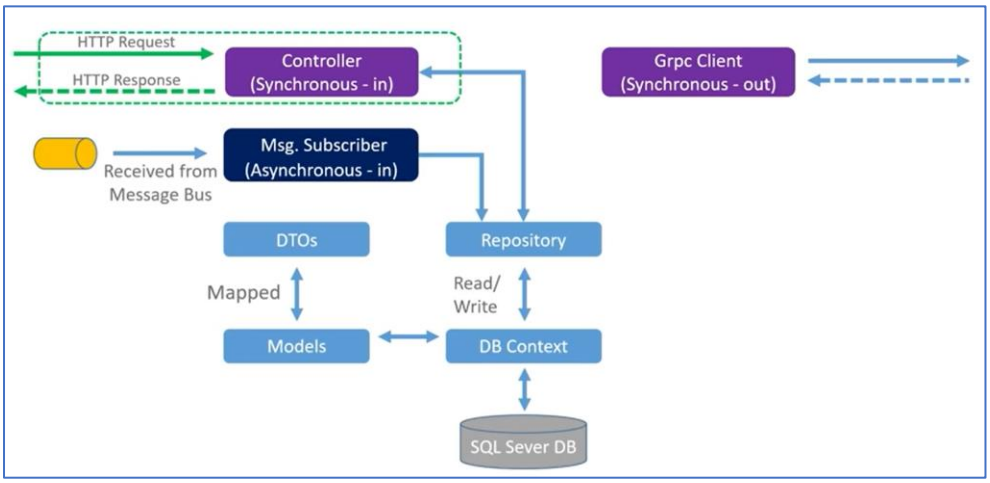
Stav Sofer 2021

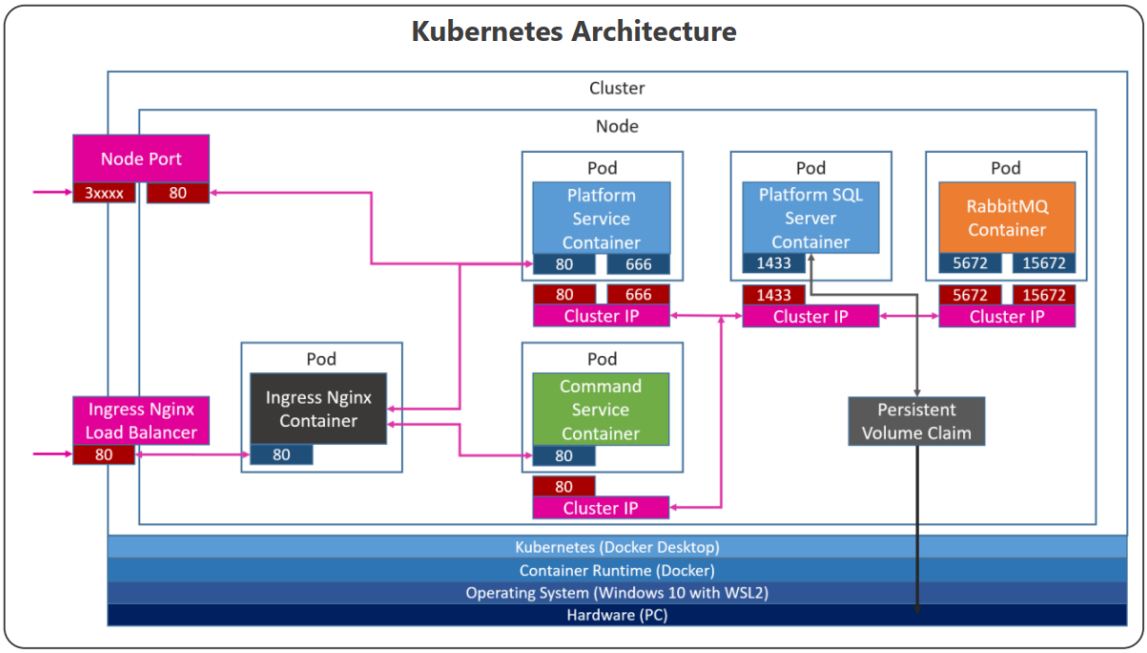


Platforms Service:



Commands Service:





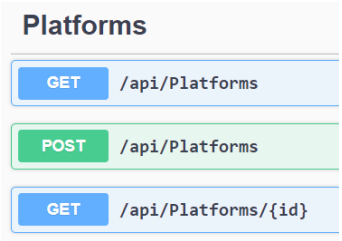
Technologies In Use:

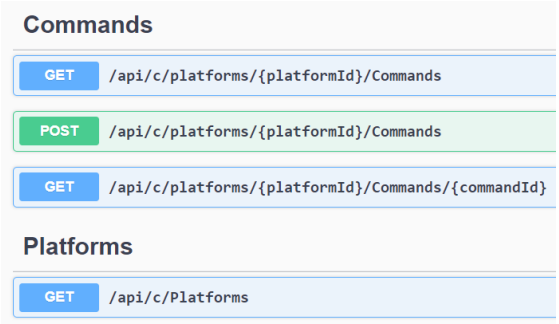
- Dotnet 5
- Microservices
- Docker
- Kubernetes
- Synchronous Http Restful API
- Synchronous gRPC & Protobufs
- Asynchronous Events Based Messaging using RabbitMQ
- Background Service
- API Gateway – Ingress Nginx
- Dependency Injection & Interface-Repository Approach
- Data Transfer Objects (DTOs)
- Auto Mapper
- Entity Framework Core

Endpoints & URIs:

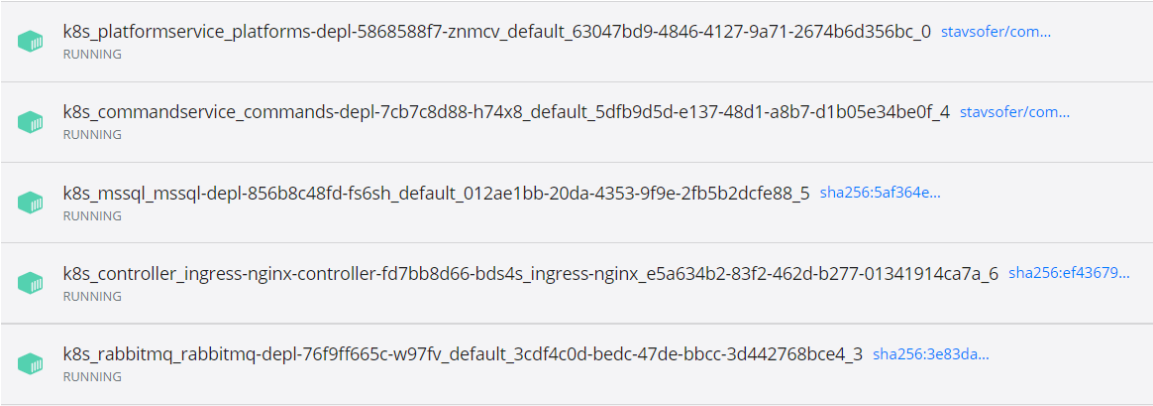
Local base URL: https://localhost:5001/ & https://localhost:6001/

Production base URL: http://acme.com/





K8S Pods on Docker Desktop:



Useful Commands:

Docker:

docker build -t <docker user id>/<image name>:<version> .
docker push <docker user id>**/**<image name>**:**<version>
docker ps
docker run -p <external port>:<internal port> -d <docker user id>/<image name>
docker stop <container Id>

docker start <container Id>

K8S:

Kubectl apply -f <name of yaml file>

Kubectl rollout restart deployment <name of deployment>

Kubectl get namespace

Kubectl get deployments
Kubectl get pods –namespace=<name of namespace>

Kubectl get <object type>
Kubectl delete <object type> <object name>

kubectl scale --replicas=<i> deployment/<name of deployment>

(i=0 → Stop the pod, i=1 → start the pod)