**Product Management Trainee Assignment – Problem Statement 3**

**1. Overview**

This technical task involves creating a GoLang-based web application that displays the current date and time, containerizing it using Docker, pushing the image to DockerHub, deploying it to a Kubernetes cluster using a declarative approach with two replicas, and finally exposing it to the internet.

**2. Step-by-Step Implementation**

Step 1: Create GoLang Program and Push to DockerHub

- Developed a simple GoLang program to serve current date and time on HTTP port.  
- Created a Dockerfile to containerize the GoLang app.  
- Built the Docker image using `docker build -t username/datetime-app .`  
- Pushed the image to DockerHub using `docker push username/datetime-app`

Step 2: Deploy to Kubernetes Using Declarative Approach

- Created a Kubernetes Deployment YAML file with 2 replicas.  
- Used `kubectl apply -f deployment.yaml` to deploy the app.  
- Verified deployment using `kubectl get pods` and `kubectl get deployments`

Step 3: Expose the App on the Internet

- Created a Service YAML file to expose the deployment using NodePort or LoadBalancer.  
- Applied the service with `kubectl apply -f service.yaml`  
- Accessed the app through public IP and exposed port to verify the output

**3. Tools and Platforms Used**

- GoLang for web app development

- Docker for containerization

- DockerHub for hosting image

- Kubernetes (Minikube/Qwiklabs/Play with K8s) for deployment

**4. Conclusion**

Successfully implemented a simple GoLang web application to show the current date and time. The application was containerized, pushed to DockerHub, deployed with 2 replicas in Kubernetes, and exposed on the internet. This project demonstrates basic understanding of GoLang, Docker, and Kubernetes using a declarative deployment strategy.