Datetime App – Dockerized Go Web Application(screenshots in the end of the page)

Legion Developed by: Danish Harrison

m Date: 23/5/2025

⋄ Objective

To create a simple web application in Go that returns the current datetime, containerize it using Docker, and run it locally on port 8080.

Project Structure

```
datetime-app/
├── main.go
├── go.mod
├── Dockerfile
└── .dockerignore
```

1 Go Application Code

main.go

```
package main

import (
          "fmt"
          "net/http"
          "time"
)

func handler(w http.ResponseWriter, r *http.Request) {
          currentTime := time.Now().Format("2006-01-02 15:04:05")
```

```
fmt.Fprintf(w, "Current DateTime: %s", currentTime)
}

func main() {
    http.HandleFunc("/", handler)
    fmt.Println("Starting server on :8080...")
    http.ListenAndServe(":8080", nil)
}
```

2 Initialize Go Module

Open terminal and run: go mod init datetime-app go mod tidy

3 Create Dockerfile

Dockerfile

FROM golang: 1.20
WORKDIR /app
COPY . .
RUN go build -o app

EXPOSE 8080 CMD ["./app"]

4 Create .dockerignore

.dockerignore

Dockerfile

*.md

5 Build Docker Image

docker build -t danish0/datetime-app:latest.

✓ This compiles the Go app and packages it in a container.

6 Run Docker Container

docker run -p 8080:8080 danish0/datetime-app:latest

Open browser and go to: http://localhost:8080You should see the current date and time printed.

Optional: Push to Docker Hub

docker login docker push danish0/datetime-app:latest

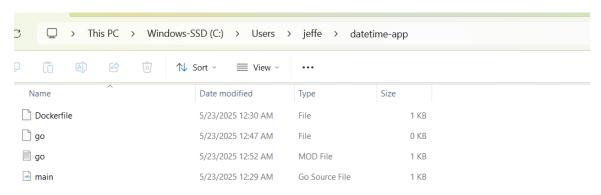
Others can run it via: docker run -p 8080:8080 danish0/datetime-app:latest

Conclusion

This project demonstrates how to:

- Write a simple web server in Go.
- Use Go modules.
- Build and run Go code in a Docker container.
- Expose services via Docker ports.

Directory:



Application output:



Current date and time: 2025-05-22 19:25:38