

Watch:  [Ejemplo con una aplicación en Go](#)

13:16 |

El código de la aplicación Go es muy sencillo:

```
package main

import (
    "flag"
    "fmt"
    "github.com/prometheus/client_golang/prometheus"
    "github.com/prometheus/client_golang/prometheus/promauto"
    "log"
    "math/rand"
    "net/http"
    "time"

    "github.com/prometheus/client_golang/prometheus/promhttp"
)

var addr = flag.String("listen-address", ":8081", "The address to listen on for HTTP requests.")

var (
    c = promauto.NewCounter(prometheus.CounterOpts{
```

```

    Name: "codely_app_sample_metric",
    Help: "Sample metric for Codely course",
})

h = promauto.NewHistogram(prometheus.HistogramOpts{
    Name: "codely_app_sample_histogram",
    Help: "Sample histogram for Codely course",
})

d = promauto.NewCounterVec(prometheus.CounterOpts{
    Name: "codely_app_sample_devices",
    Help: "Sample counter opts devices for Codely course"},
[]string{"device"})

func main() {

    go func() {
        for {
            rand.Seed(time.Now().UnixNano())
            h.Observe(float64(rand.Intn(100-0+1) + 0))
            d.With(prometheus.Labels{"device":"/dev/sda"}).Inc()
            c.Inc()
            fmt.Print(".")
            time.Sleep(1 * time.Second)
        }
    }()

    http.Handle("/metrics", promhttp.Handler())
    log.Fatal(http.ListenAndServe(*addr, nil))
}

```

Levantamos nuestra infraestructura, y además, con la ayuda de un Dockerfile, nuestra app en un nuevo contenedor.

El Dockerfile podría es:

```

FROM golang:latest AS builder
ADD ./app /app
WORKDIR /app
RUN go mod download
RUN CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build -a -o /main
.

FROM alpine:latest
RUN apk --no-cache add ca-certificates
COPY --from=builder /main ./

```

```
RUN chmod +x ./main
ENTRYPOINT ["./main"]
EXPOSE 80
```

Por tanto, el archivo de configuración para docker-compose sería:

```
version: '3.7'
services:

  prometheus:
    image: prom/prometheus:v2.1.0
    volumes:
      - ./etc/prometheus:/etc/prometheus/
      - ./var/prometheus/prometheus_data:/prometheus
    command:
      - '--config.file=/etc/prometheus/prometheus.yml'
      - '--storage.tsdb.path=/prometheus'
      - '--
web.console.libraries=/usr/share/prometheus/console_libraries'
      - '--
web.console.templates=/usr/share/prometheus/consoles'
    ports:
      - 9090:9090
    depends_on:
      - cadvisor

  app:
    build: .
    restart: always
    ports:
      - 8081:8081
    depends_on:
      - prometheus
```

Y el archivo de config de Prometheus:

```
global:
  scrape_interval: 15s
  evaluation_interval: 15s
  external_labels:
    monitor: 'codely-app'

scrape_configs:
```

```
- job_name: 'prometheus'
  scrape_interval: 5s
  static_configs:
    - targets: ['localhost:9090']

- job_name: 'codely'
  scrape_interval: 5s
  static_configs:
    - targets: ['app:8081']
```