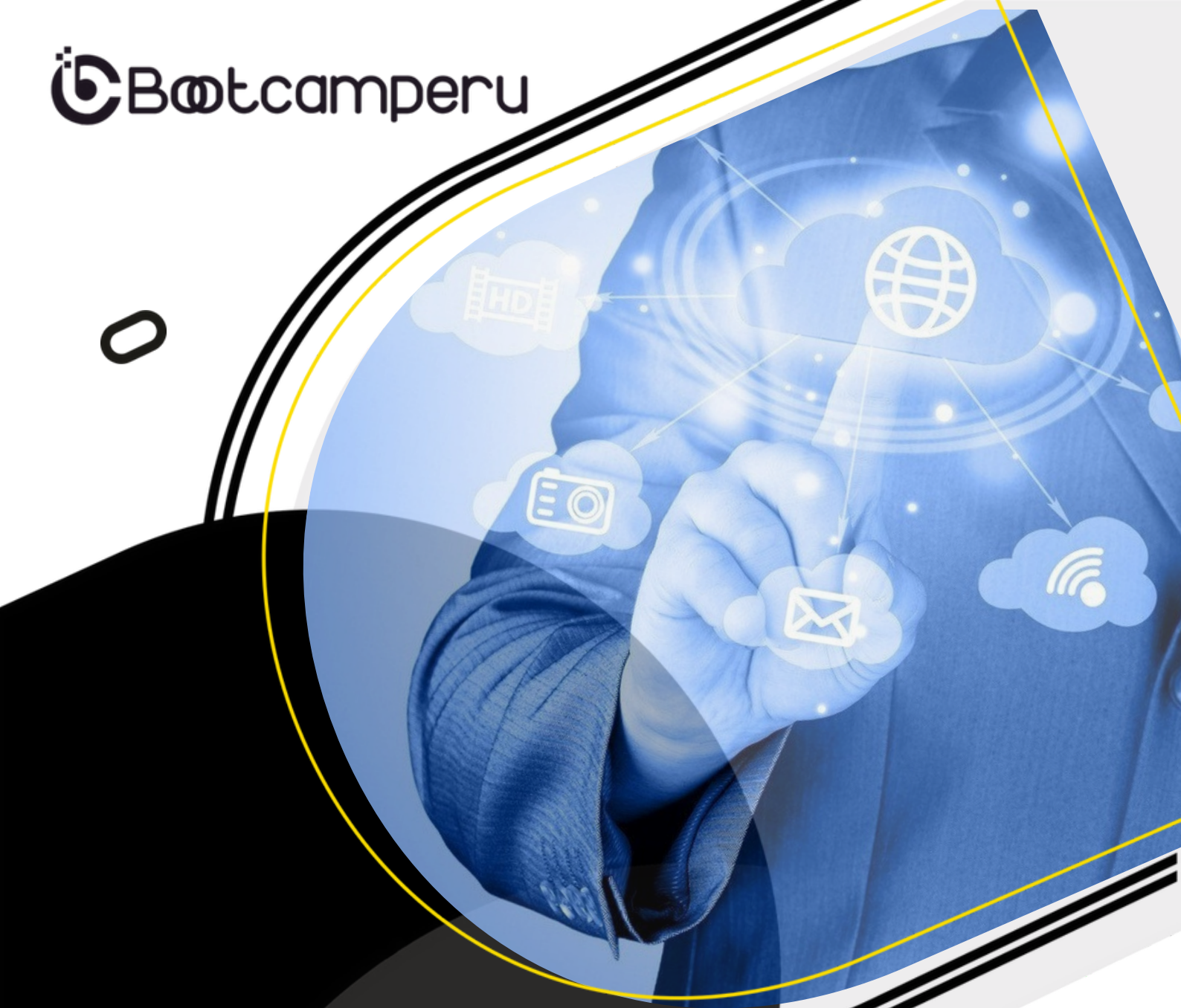




BOOTCAMP

**Observabilidad desde Cero
con Prometheus y Grafana**

Objetivos del Curso

- **Comprender los fundamentos del monitoreo y la observabilidad**, incluyendo telemetría, métricas, logs, trazas, indicadores SLI/SLO/SLA y la arquitectura de una plataforma de observabilidad.
- **Instalar y configurar Prometheus y Grafana** para recopilar métricas de infraestructura, contenedores y aplicaciones mediante exporters y scraping de endpoints.
- **Crear consultas con PromQL**, incluyendo selectores, operadores, funciones de agregación, rate, irate, histogram_quantile, recording rules y optimización de consultas.
- **Monitorear servidores Linux y contenedores Docker**, utilizando Node Exporter, cAdvisor y Blackbox Exporter para métricas de CPU, memoria, disco, red y disponibilidad.
- **Construir dashboards profesionales en Grafana**, aplicando paneles, variables dinámicas, transformaciones, anotaciones y buenas prácticas de visualización.
- **Configurar alertas y gestión de incidentes con Alertmanager**, incluyendo reglas de alerta, rutas, receptores, silencios, Grafana Alerting y elaboración de runbooks.



Horarios

INICIO DE CLASES:

Sábado, 23 de Julio 2026

Horario (Hora Perú):

Jueves : 7:30pm - 10:00pm

Martes : 7:30pm - 10:00pm



Temario

DURACIÓN TOTAL: CLASES EN VIVO (60 HORAS ACADÉMICAS)

Modalidad: Online en vivo 

Duración total: 3 MESES

Nivel: Básico a intermedio

Requisitos: Conocimientos básicos de informática y terminal.

M1

Módulo 1: Fundamentos de monitoreo y observabilidad

- ¿Qué es el monitoreo y la observabilidad?
- Diferencias entre monitoreo tradicional y observabilidad
- Telemetría: métricas, logs y trazas
- Disponibilidad, rendimiento y confiabilidad
- Latencia, tráfico, errores y saturación
- Introducción a indicadores SLI, SLO y SLA
- Casos de uso de observabilidad
- Flujo de detección y análisis de incidentes
- Arquitectura general de una plataforma de observabilidad

Práctica: Análisis de un incidente y definición de métricas necesarias

M2

Módulo 2: Preparación del entorno con Linux y Docker

- Conceptos básicos de Linux
- Procesos, servicios, puertos y recursos del sistema
- Comandos básicos para diagnóstico
- Introducción a archivos YAML
- Fundamentos de contenedores
- Instalación y configuración de Docker
- Redes, volúmenes y puertos en Docker
- Introducción a Docker Compose
- Creación del laboratorio del curso

Práctica: Levantar Prometheus y Grafana mediante Docker Compose





M3

Módulo 3: Fundamentos y arquitectura de Prometheus

- ¿Qué es Prometheus?
- Componentes principales: Server, TSDB
- Targets, jobs e instances
- Funcionamiento del proceso de scraping
- Endpoints /metrics
- Configuración del archivo prometheus.yml
- Intervalos de recopilación y evaluación
- Descubrimiento estático de servicios
- Validación con Promtool
- Estado y diagnóstico de targets

Práctica: Configurar diferentes targets y verificar el estado de cada servicio

M4

Módulo 4: Métricas, exporters e instrumentación

- Concepto de métrica: nombre, valor, timestamp y etiquetas
- Contadores, Gauges, Histogramas y Summaries
- Diferencias y casos de uso
- Buenas prácticas para nombrar métricas
- Cardinalidad y sus riesgos
- Introducción a exporters y Node Exporter
- Métricas personalizadas
- Instrumentación básica de una aplicación Python
- Exposición de métricas mediante endpoint HTTP

- Práctica: Crear métricas personalizadas para solicitudes, errores y tiempo de respuesta**

M5

Módulo 5: PromQL desde cero

- Introducción a PromQL
- Selectores instantáneos y de rango
- Filtros mediante etiquetas
- Operadores aritméticos, de comparación y lógicos
- Funciones básicas
- Agregaciones: sum, avg, min, max, count
- Agrupaciones con by y without
- Uso de rate, irate e increase
- Cálculo de consumo de CPU, memoria y red
- Práctica: Construir consultas para identificar consumo elevado y errores**



M6**Módulo 6: PromQL aplicado y optimización de consultas**

- Combinación de métricas y vector matching
- Filtros avanzados y cálculo de porcentajes
- Consultas para disponibilidad, latencia y tasas de error
- Trabajo con histogramas
- Percentiles P90, P95 y P99 con histogram_quantile
- Detección de métricas ausentes
- Introducción a recording rules
- Optimización y reutilización de consultas
- Diagnóstico de consultas lentas
- Errores frecuentes en PromQL

Práctica: Crear consultas para evaluar la salud de una aplicación web

M7**Módulo 7: Monitoreo de infraestructura y contenedores**

- Instalación y configuración de Node Exporter
- Monitoreo de CPU, memoria, disco y red
- Carga del sistema, procesos y disponibilidad
- Introducción al monitoreo de contenedores
- Instalación de cAdvisor
- Consumo de CPU y memoria por contenedor
- Estado y reinicio de contenedores
- Monitoreo de aplicaciones web
- Pruebas de disponibilidad con Blackbox Exporter

Práctica: Monitorear un servidor Linux y varios contenedores desde Prometheus

M8**Módulo 8: Visualización y dashboards con Grafana**

- Arquitectura y funcionamiento de Grafana
- Configuración de Prometheus como fuente de datos
- Creación de dashboards y paneles
- Series de tiempo, Gauge, Stat panels, tablas
- Configuración de unidades y umbrales
- Variables dinámicas y filtros
- Transformaciones de datos y anotaciones
- Importación/exportación de dashboards
- Buenas prácticas de diseño

Práctica: Crear dashboard de infraestructura y otro para aplicación web

Módulo 9: Alertas y gestión de incidentes

Fundamentos de alertamiento
Reglas de alerta en Prometheus
Configuración de expresiones PromQL
Duración, evaluación, severidad y etiquetas
Anotaciones y mensajes
Introducción a Alertmanager
Agrupación y deduplicación de alertas
Rutas, receptores, silencios e inhibiciones
Grafana Alerting y contact points
Políticas de notificación
Prevención de fatiga de alertas
Elaboración de runbooks

Práctica: Configurar alertas por servidor caído, CPU elevado, disco y errores

Módulo 10: Implementación del proyecto final

Diseño de la solución de observabilidad
Creación de la arquitectura
Instalación mediante Docker Compose
Configuración de Prometheus y exporters
Instrumentación de la aplicación
Construcción de consultas PromQL
Creación de dashboards
Configuración de alertas
Simulación de incidentes
Diagnóstico y resolución de problemas
Seguridad básica de Prometheus y Grafana
Respaldo de configuraciones y dashboards
Documentación técnica
Presentación y sustentación del proyecto

Proyecto Final

Metodología y Evaluación

Metodología: 25% explicación conceptual, 75% laboratorios y proyecto. Demostraciones en vivo, prácticas guiadas, casos reales de monitoreo, simulación de incidentes y desarrollo progresivo del proyecto final.

Evaluación:

- Laboratorios y ejercicios: 40%
- Retos de PromQL y dashboards: 20%
- Proyecto final: 40%

Entregables: docker-compose.yml, prometheus.yml, reglas de alertas, aplicación instrumentada, dashboards de infraestructura y aplicación, configuración de notificaciones, repositorio en GitHub, documento de instalación y runbook.