

BOOTCAMP

**Full Stack Engineering,
Arquitectura de Software y Cloud Native**

Objetivos del Programa

Diseñar arquitecturas Full Stack escalables, mantenibles y orientadas a entornos empresariales.

Construir aplicaciones frontend avanzadas utilizando React, TypeScript y arquitecturas modulares.

Desarrollar servicios backend con FastAPI, aplicando Clean Architecture, Domain-Driven Design y patrones de diseño.

Diseñar soluciones basadas en microservicios, comunicación asíncrona y arquitectura orientada a eventos.

Implementar seguridad, pruebas automatizadas, observabilidad, integración continua y despliegue cloud.

Desplegar soluciones productivas mediante Docker, Kubernetes, Terraform y servicios administrados de nube.

Aplicar IA generativa como apoyo en arquitectura, desarrollo, documentación, pruebas y análisis de código.



Horarios

INICIO DE CLASES:

Jueves, 30 de Julio 2026

Horario (Hora Perú):

Martes y Jueves : 7:30 p.m. – 10:00pm

Temario

DURACIÓN TOTAL : CLASES EN VIVO (240 HORAS ACADÉMICAS)

Modalidad: Online en vivo 

Duración total : 6 MESES

Nivel: Avanzado

Requisitos: El participante deberá contar con:

- Conocimientos de HTML, CSS y JavaScript moderno.
- Experiencia básica o intermedia con React.
- Conocimientos de Python y FastAPI.
- Manejo de APIs REST.
- Conocimientos de SQL y bases de datos.
- Manejo básico de Git y GitHub.
- Fundamentos de Docker y despliegue de aplicaciones.
- Haber culminado el Bootcamp Full Stack Web Developer de Bootcamperu o demostrar conocimientos equivalentes.

Dirigido a quienes:

- Ya desarrollan aplicaciones.
- Conocen frontend y backend.
- Buscan arquitectura, cloud, microservicios, seguridad y producción.

M1

Módulo 1: Ingeniería de Software y Arquitectura de Aplicaciones

- 🔗 Evolución de una aplicación monolítica hacia una arquitectura empresarial
- 🔗 Arquitectura en capas, hexagonal y Clean Architecture
- 🔗 Principios SOLID, DRY, KISS y separación de responsabilidades
- 🔗 Patrones de diseño aplicados a frontend y backend
- 🔗 Diseño orientado al dominio: introducción a DDD
- 🔗 Entidades, objetos de valor, servicios y repositorios
- 🔗 Diagramas de arquitectura con C4 Model
- 🔗 Architecture Decision Records
- 🔗 Gestión profesional de proyectos con GitHub



M2 Módulo 2: Frontend Engineering con React y TypeScript

- TypeScript aplicado a aplicaciones React
- Arquitectura frontend basada en features
- Componentes reutilizables y principios de composición
- Custom Hooks y manejo avanzado de estado con Redux Toolkit
- Formularios con React Hook Form y validaciones con Zod
- Autenticación y autorización basada en roles
- Lazy loading, code splitting y Suspense



M3 Módulo 3: Diseño de Sistemas Frontend, UX y Rendimiento

- Diseño de sistemas y librerías de componentes
- Design tokens y Storybook
- Accesibilidad avanzada e internacionalización
- Gestión de temas y personalización
- Core Web Vitals y optimización del rendimiento
- Progressive Web Apps
- Error Boundaries y monitoreo de errores
- Microfrontends: fundamentos y casos de uso

M4 Módulo 4: Backend Engineering Avanzado con Python y FastAPI

- Programación asíncrona avanzada con Python
- Arquitectura modular con FastAPI y Clean Architecture
- Inyección de dependencias
- Repository Pattern y Unit of Work
- Manejo global de excepciones
- Validación avanzada con Pydantic
- WebSockets y comunicación en tiempo real
- Optimización y profiling de aplicaciones Python

M5 Módulo 5: Bases de Datos, Caché y Persistencia Avanzada

- Modelado de datos para sistemas empresariales
- PostgreSQL avanzado: índices y optimización
- Transacciones y niveles de aislamiento
- Migraciones con Alembic
- SQLAlchemy y SQLAlchemyModel avanzado
- MongoDB para documentos no estructurados
- Redis para caché, sesiones y datos temporales
- Backup, recuperación y replicación

M6 Módulo 6: Microservicios y Arquitecturas Orientadas a Eventos

- Monolito modular frente a microservicios
- Comunicación síncrona y asíncrona
- Mensajería con RabbitMQ o Apache Kafka
- Arquitectura Event-Driven
- Patrón Saga y Outbox Pattern
- Circuit Breaker, Retry y Timeout
- API Gateway y Service Discovery
- Introducción a CQRS

M7 Módulo 7: Seguridad de Aplicaciones Full Stack

- OWASP Top 10 y seguridad por diseño
- Autenticación avanzada con JWT y OAuth 2.0
- Autorización mediante roles y permisos (RBAC)
- Protección contra CSRF, XSS, SQL Injection
- Rate limiting y gestión de secretos
- Seguridad en APIs y microservicios
- Seguridad de imágenes Docker
- Auditoría y registro de eventos de seguridad

M8 Módulo 8: Testing, Calidad y Automatización

- Pruebas unitarias con Pytest
- Pruebas de integración y APIs
- Testcontainers
- Pruebas de componentes React con Testing Library
- Pruebas end-to-end con Playwright
- Pruebas de carga con Locust o k6
- SonarQube y métricas de calidad
- IA aplicada a generación y revisión de pruebas

M9 Módulo 9: DevOps, CI/CD e Infraestructura como Código

- Docker avanzado y multi-stage builds
- Docker Compose para entornos distribuidos
- GitHub Actions, GitLab CI o Jenkins
- Estrategias Blue-Green y Canary
- Kubernetes avanzado y Helm Charts
- Infraestructura como código con Terraform
- Gestión de ambientes dev, test y producción

M10 Módulo 10: Cloud Architecture y Servicios Administrados

- 🕒 Diseño de arquitecturas cloud
- 🕒 Alta disponibilidad y tolerancia a fallos
- 🕒 Escalabilidad horizontal y vertical
- 🕒 Despliegue de aplicaciones en AWS
- 🕒 Contenedores con ECS o EKS
- 🕒 Servicios equivalentes en Azure y GCP
- 🕒 Serverless: fundamentos y casos de uso
- 🕒 Optimización de costos e introducción a FinOps

M11 Módulo 11: Observabilidad, SRE y Operación en Producción

- 🕒 Logs estructurados y métricas de aplicaciones
- 🕒 Trazabilidad distribuida con OpenTelemetry
- 🕒 Prometheus y Grafana
- 🕒 Service Level Indicators y Objectives
- 🕒 Gestión de incidentes y postmortems
- 🕒 Chaos Engineering: introducción
- 🕒 Optimización de rendimiento en producción

M12 Módulo 12: Proyecto Final - Plataforma Full Stack Cloud Native

- 🕒 Frontend con React y TypeScript, arquitectura modular
- 🕒 Backend con Python y FastAPI, Clean Architecture
- 🕒 PostgreSQL, MongoDB y Redis según el caso
- 🕒 Microservicios o monolito modular justificado
- 🕒 Pruebas unitarias, integración y end-to-end
- 🕒 Dockerización y pipeline CI/CD
- 🕒 Despliegue en AWS, Azure o GCP con Kubernetes
- 🕒 Monitoreo, logs y alertas
- Documentación técnica y defensa del proyecto