SYLLABUS

SOFTWARE DEVELOPMENT · COURSE HANDOUT 2026

MICROSERVICES

UNDARIES . CONTRACTS . DA
GATEWAY . DISCOVERY . RES
NTS . TRACING . DEPLOYMEN

DURATION	LEVEL	MODULES	MODE
60 Hours	**Advanced**	**8**	**Classroom · Online · Hybrid**

01 COURSE OVERVIEW

A microservices course on decomposition, inter-service communication and operating a distributed system. The course is honest about cost: it starts with when not to split a system, then covers domain boundaries, data ownership, synchronous and event-driven communication, resilience patterns, and the observability required to run services in production.

Learners decompose a monolith into three services with a gateway, message broker and distributed tracing.

WHO IT IS FOR

- Backend and platform engineers
- Architects planning decomposition
- DevOps engineers supporting service estates
- Technical leads

PREREQUISITES

Strong backend development, Docker and REST API experience.

02 LEARNING OUTCOMES

01 Decide whether a system should be split at all.

02 Define service boundaries around business capability.

03 Own data per service and handle eventual consistency.

04 Communicate synchronously and via events.

05 Apply retry, timeout, circuit breaker and bulkhead patterns.

06 Trace requests across services.

07 Deploy and version services independently.

03 MODULE STRUCTURE

8 MODULES

01 FOUNDATIONS

- Monolith vs microservices
- Conway's law
- Bounded contexts
- Costs and prerequisites
- Domain-driven design basics

02 SERVICE DESIGN

- Capability boundaries
- Versioning
- Service templates
- API contracts
- Shared libraries and their risks

03 DATA MANAGEMENT

- Database per service
- Saga pattern
- CQRS
- Eventual consistency
- Outbox pattern
- Reporting across services

LAB

SAGA IMPLEMENTATION

04 SYNCHRONOUS COMMUNICATION

- REST between services
- API gateway
- Load balancing
- gRPC
- Service discovery
- Timeouts

LAB

GATEWAY AND DISCOVERY

05 EVENT-DRIVEN COMMUNICATION

- Message brokers
- Topics and consumer groups
- Idempotent consumers
- Kafka and RabbitMQ basics
- Ordering

LAB

EVENT PIPELINE

06 RESILIENCE

- Retries with backoff
- Bulkheads
- Graceful degradation
- Circuit breakers
- Rate limiting
- Chaos testing basics

07 OBSERVABILITY

- Structured logging
- Distributed tracing
- SLOs and alerting
- Metrics
- Correlation IDs

TOOLS

OPENTELEMETRY · PROMETHEUS · GRAFANA · JAEGER

08 DEPLOYMENT AND GOVERNANCE

- Containers
- Independent pipelines
- Service catalogue
- Kubernetes deployment
- Contract testing
- Ownership

PROJECT

DECOMPOSED THREE-SERVICE SYSTEM

04 ASSESSMENT & CERTIFICATION

Module assignments and labs	**25%**
Mini projects	**20%**
Internal assessments	**15%**
Final project and review	**40%**

CERTIFICATION

Learners who complete all modules, submit the final project and clear the review receive a course completion certificate from Kaizen Infinities Private Limited. Project work is documented for the learner's portfolio, and interview preparation is included in the closing sessions.

05 CAREER OUTCOMES

ROLES THIS PROGRAMME PREPARES FOR

MICROSERVICES DEVELOPER · BACKEND ARCHITECT · PLATFORM ENGINEER · DISTRIBUTED SYSTEMS ENGINEER · TECHNICAL LEAD

CALL US
+91 73051 33996

EMAIL US
apply@kaizeninfinities.com

VISIT OUR WEBSITE
www.kaizeninfinities.com