



# OpenTracing Foundation Certification

## About DevOpsSchool

DevOpsSchool is a unit of "Cotocus PVT Ltd" and a leading platform which helps IT organizations and professionals to learn all the emerging technologies and trend which helps them to learn and embrace all the skills, intelligence, innovation and transformation which requires to achieve the end result, quickly and efficiently. We provide over 40 specialized programs on DevOps, Cloud, Containers, Security, AI, ML and on Big data that are focused on industry requirement and each curriculum is developed and delivered by leading experts in each domain and aligned with the industry standards.

## About Course

Microservices are loosely coupled independent services, that work in tandem to execute a function or a request. The communication between individual services happen via interconnected APIs. Since each service works independently, dependencies between services are reduced and this favours scalability and faster code deployments.

But complex applications tend to bring complex problems. In a microservice architecture, when an error occurs, it is difficult to trace back to the root cause of the error because of the interconnected services. So rather than looking at how to resolve the error, finding the error becomes a challenge in itself. And it wont help to just look at the root cause without enough context. Since every service is independent on it's own, correlating the logs, metrics, and traces of the involved services and narrowing down to the exact issue may take time and effort, more than anticipated. This is where distributed tracing comes handy.

Distributed Tracing is the process of tracking and analyzing what happens to a request (transaction) across all services it touches. Distributed tracing describes the act of following a transaction through all participating applications (tiers) and sub-systems, such as databases.

Using any of these tools and their specific components results in vendor lock-in, making it harder for developers to work because not all vendors have the same support across different frameworks and libraries. To solve this problem, OpenTracing and OpenCensus projects were started. They provide what other frameworks and libraries can implement. This enables developers to add instrumentation to their application code that won't lock them into any particular vendor. This low coupling, along with easy-to-use Interfaces, makes these two projects very attractive.



Co-coordinator - Akanksha Kumari

Call/WhatsApp: - +91 1800 889 7977

Mail Address: -

[contact@DevOpsSchool.com](mailto:contact@DevOpsSchool.com)

Secondary contact - Patrick

Call/WhatsApp: - +91 7004 215 841

Mail Address: - [contact@DevOpsSchool.com](mailto:contact@DevOpsSchool.com)

Duration	14 Hours
Mode	Online (Instructor-led, live & Interactive)
Projects (Real time scenario based)	1

FEATURES	DEVOPSSCHOOL	OTHERS
Faculty Profile Check	✓	✗
Lifetime Technical Support	✓	✗
Lifetime LMS access	✓	✗
Top 25 Tools	✓	✗
Interviews Kit	✓	✗
Training Notes	✓	✗
Step by Step Web Based Tutorials	✓	✗
Training Slides	✓	✗
Training + Additional Videos	✓	✗



## Projects

---

As part of this OpenTracing Foundation Certification, you will be working on real-time projects and assignments that have immense implications in the real-world industry scenarios, thus helping you fast track your career effortlessly. At the end of this training program, there will be a quiz that perfectly reflects the type of questions asked in the certification exams and helps you score better marks.

## Interview

---

As part of this, You would be given complete interview preparations kit, set to be ready for the OpenTracing Foundation hotseat. This kit has been crafted by 200+ years industry experience and the experiences of nearly 10000 DevOpsSchool OpenTracing Foundation learners worldwide..

# AGENDA OF THE OPENTRACING FOUNDATION CERTIFICATION

---

## Introduction

---

Networking and observability in distributed systems

---

The growing importance of distributed tracing instrumentation.

---

## The Problem with Existing Tracing Solutions

---

Communication package maintenance, inconsistencies across platforms, etc.

---

## How OpenTracing Standardizes Distributed Tracing

---

## Setting up an OpenTracing Software

---

- Jaeger
- Zipkin

## Overview of OpenTracing Features and Architecture

---

## OpenTracing API: Tracer and Span

---

## The Tracing Process

---

Services, packages, application-specific business logic.

---

Transaction destiny, reference, type, timestamps, etc.

---

A 'Hello World' Example

---

Identifying a Misbehaving System

---

Analyzing Tags, Spans, Scope, Threads, and Logs

---

Determining "how" an error occurred

---

Debugging a Distributed System

---

Filtering performance monitoring

---

Analyzing service dependencies

---

Root cause analysis

---

Optimizing code

---

Configuring Environment Variables

---

Adjusting frequency of events

---

Case Study: Ensuring Compliance in Legacy Systems

---

Monitoring Live Environments

---

Integrating OpenTracing with RPC and other frameworks

---

Server-side and client-side tracing

---

Best Practices

---

Troubleshooting

---

Summary and Conclusion

---





# Thank you!

Connect with us for more info

Call/WhatsApp: - +91 968 682 9970

Mail: - [contact@DevOpsSchool.com](mailto:contact@DevOpsSchool.com)

[www.DevOpsSchool.com](http://www.DevOpsSchool.com)