

## CORSO PROGRAMMAZIONE ORIENTATA AI MICROSERVIZI E CYBERSECURITY

### OBIETTIVI DEL CORSO

Offrire la possibilità ai giovani informatici di **acquire skills nello sviluppo di software** e adottare uno stile architetturale a microservizi capace di evolversi con rapidità e continuità nel tempo con un occhio aperto alla **cyber security** e alle **vulnerabilità del codice**. Questo corso consentirà agli informatici di domani di acquisire le principali tecniche ed approcci di sviluppo centralizzato e distribuito, tenendo conto delle minacce legate al cyberspazio in cui operano, ampliando così la propria "cassetta degli attrezzi".

### DESTINATARI E DURATA DEL CORSO

Il corso è destinato ai giovani diplomati in Informatica e studenti iscritti al I e II anno del CdL in Informatica.

### COMPETENZE

- Lavorare in team tecnici
- Comprendere la struttura di un sito web e sapere cosa ti serve per realizzarla (server, database, backend...)
- Gestire le attività Back-End di siti/applicazioni

### ARTICOLAZIONE E STRUTTURA DEL PERCORSO FORMATIVO

Percorso formativo, della durata complessiva di 320 ore, è strutturato in tredici moduli e riguarderà tre macro-contenuti: **Microservizi-IoT, Java, Cybersecurity**

Il corso prevede un approccio **mixed methods** con:

- **280 h** di formazione frontale in aula in cui i partecipanti potranno incrementare il personale know-how informatico grazie all'interazione **face to face** sia con i propri pari che con il docente.
- **40 h** di formazione saranno svolte mediante la metodologia **Training on the Job** in cui si avvierà un percorso di apprendimento sul campo che consentirà ai partecipanti di apprendere dall'esperienza concreta.

## PROGRAMMAZIONE ORIENTATA AI MICROSERVIZI E CYBERSECURITY

MODULI	CORSO	CONTENUTI DEL CORSO
<b>Mod. 1</b>	<b>FONDAMENTI DI OOP</b> (8h)	<ul style="list-style-type: none"> <li>• OOP (object-oriented programming)</li> <li>• Cenni di Progettazione OO</li> <li>• Modellazione Funzionale</li> </ul>
<b>Mod. 2</b>	<b>JAVA</b> (40h)	<ul style="list-style-type: none"> <li>• I costrutti del linguaggio Java</li> <li>• Componenti fondamentali di un programma</li> <li>• Identificatori e tipi di dati</li> <li>• Operatori e Costrutti</li> <li>• Java I/O</li> </ul>
<b>Mod. 3</b>	<b>USO E CARATTERISTICHE DELLE PRINCIPALI STRUTTURE DATI</b> (12h)	<ul style="list-style-type: none"> <li>• Cenni sulle strutture dati (liste e vettori)</li> <li>• Cenni sugli algoritmi</li> <li>• Ricerca di un elemento in una lista e in un vettore</li> <li>• Algoritmi di ordinamento</li> </ul>
<b>Mod. 4</b>	<b>DATABASE RELAZIONALI</b> (28h)	<ul style="list-style-type: none"> <li>• Introduzione al linguaggio SQL (Structured Query Language)</li> <li>• Linguaggio SQL: Differenza tra DDL e DML</li> <li>• Caratteristiche del DDL Data Definition Language</li> <li>• Caratteristiche del DML (Data Manipulation Language)</li> <li>• Leggere, inserire e modificare i dati in un database mediante DML</li> </ul>
<b>Mod. 5</b>	<b>INTERFACCIAMENTO CON LO STRATO DI PERSISTENZA</b> (10h)	<ul style="list-style-type: none"> <li>• Persistenza Applicazioni Enterprise: Uso dei modelli</li> <li>• Il problema della persistenza</li> <li>• I driver JDBC</li> <li>• Modalità di accesso con JBDC</li> </ul>
<b>Mod. 6</b>	<b>TRACING E LOGGING</b> (6h)	<ul style="list-style-type: none"> <li>• Modalità di simulazione dell'andamento dell'algoritmo per verificare la sua correttezza: Tracing</li> <li>• Modalità di mappatura dei dati</li> <li>• Principali caratteristiche della utility open source di logging scritta in Java: Log4J</li> </ul>
<b>Mod. 7</b>	<b>DESIGN PATTERN</b> (12h)	<ul style="list-style-type: none"> <li>• Introduzione alle microarchitetture</li> <li>• Principi generali di una buona progettazione basata su Design Pattern</li> <li>• Utilizzo dei Design Pattern in Java</li> </ul>
<b>Mod. 8</b>	<b>CENNI DI PROGRAMMAZIONE DISTRIBUITA (MQTT E REST)</b> (12h)	<ul style="list-style-type: none"> <li>• Elementi di programmazione distribuita in Java</li> <li>• Cenni su MQTT</li> <li>• Cenni sulle REST API</li> <li>• Sviluppo di servizi REST</li> </ul>
<b>Mod. 9</b>	<b>CENNI DI SOFTWARE TESTING</b> (24h)	<ul style="list-style-type: none"> <li>• Introduzione al processo di testing</li> <li>• Testing di Unità</li> <li>• Le 3 A dell'unit testing: Arrange – Act – Assert</li> <li>• JUnit come strumento di test automation</li> <li>• Testing di regressione</li> </ul>

<b>Mod. 10</b>	<b>CENNI DI SPRING TECHNOLOGY</b> (86h)	<ul style="list-style-type: none"> <li>• MVC</li> <li>• Una panoramica sulla Spring Technology</li> <li>• Spring MVC</li> <li>• Spring Boot</li> <li>• Il pattern architetturale per la gestione della persistenza: DAO (Data Access Object)</li> <li>• Spring Security</li> </ul>
<b>Mod. 11</b>	<b>CENNI DI OBJECT-RELATIONAL MAPPING</b> (24h)	<ul style="list-style-type: none"> <li>• Il paradigma di programmazione Object-relational mapping (ORM)</li> <li>• Cenni su Hibernate</li> <li>• L'automazione della persistenza degli oggetti mediante Hibernate</li> <li>• Tecniche di rappresentazione degli oggetti mediante grafi interconnessi con altri oggetti</li> <li>• Benefici di una soluzione ORM</li> </ul>
<b>Mod. 12</b>	<b>ECLIPSE</b> (10h)	<ul style="list-style-type: none"> <li>• Introduzione all'IDE Eclipse</li> <li>• Architettura a Plug-In ed ambiente di sviluppo</li> <li>• Debugging</li> </ul>
<b>Mod. 13</b>	<b>TOOL DI ANALISI STATICA PER LA SICUREZZA</b> (8h)	<ul style="list-style-type: none"> <li>• Cosa sono i tool di analisi statica;</li> <li>• Introduzione a SonarQube e SonarLint</li> <li>• Identificare e rimuovere vulnerabilità</li> </ul>