

FICHA TÉCNICA

INFORMACIÓN GENERAL

Código	:	XYZ-9999
Pilar	:	Desarrollo de Software
Programa	:	Backend Developer I
Certificación	:	Backend Developer I
Créditos	:	99
Duración	:	160 horas

DESCRIPCIÓN DEL CURSO

En este curso, el participante obtendrá los conocimientos y habilidades necesarias para el diseño y construcción de servicios backend con los protocolos HTTP/2, SOAP, AMQP, WebSockets y Streaming con uno de los frameworks más demandados en el mercado tecnológico. Además, diseñaremos soluciones reactivas para aprovechar los múltiples hilos (conurrencia) que posee nuestro nodo computacional, pues es la demanda de la próxima generación de software empresarial.

PÚBLICO OBJETIVO

Desarrolladores, Arquitectos de Software, Consultores y Profesionales interesados en profundizar sus conocimientos en arquitecturas distribuidas altamente disponibles.

LOGROS DEL CURSO

Al finalizar el curso, el alumno será capaz de diseñar aplicaciones altamente escalables, seguras y eficientes. Adicionalmente, estará preparado para:

1. Conocer los patrones y principios de diseño de software más relevantes
2. Construir piezas de software con el protocolo HTTP/REST con Spring WebMVC, Spring Data y Spring Boot
3. Construir piezas de software con el protocolo SOAP con Spring Web Service, Spring Data y Spring Boot
4. Construir piezas de software con el protocolo AMQP con Spring AMQP y RabbitMQ. Además implementaremos el patrón Publish/Subscriber con Spring Cloud Stream y Azure Service Bus.
5. Construir piezas de software con acceso a fuentes de datos Azure SQL, CosmosDB y Azure Redis con drivers sincrónicos y asíncrónicos
6. Construir piezas de software non-blocking con el protocolo HTTP/REST con Spring WebFlux, Reactor, Spring Data y Spring Boot
7. Soporte del lado del cliente para acceder, almacenar y revocar secretos con Azure Key Vault y Spring Cloud Azure
8. Implementar Autenticación y Autorización con Spring Security, OAuth2 y JWT
9. Realizar un servicio backend completo utilizando los diferentes componentes que provee el framework

PRE REQUISITOS

- Conocimiento de programación orientada a objetos con Java (nivel intermedio)
- Conocimiento de los principios de diseño de software SOLID
- Conocimiento de Spring Core

CAPACITACIONES RECOMENDADAS

- Java 10 Web Developer

LÍNEA DE ESPECIALIZACIÓN

CERTIFICACIONES

1. Al término de la capacitación, el alumno obtendrá un certificado de Backend Developer Senior por aprobar el programa.
2. Al completar satisfactoriamente 2 cursos de la línea de especialización (ver punto VI), se le acreditará con una Especialización en Desarrollo de Aplicaciones.

METODOLOGÍA

El enfoque práctico-reflexivo de cada una de las sesiones y la participación activa a lo largo de estas favorecerá no solo el aprendizaje de contenidos, sino también, su aplicación en la resolución de situaciones reales en diferentes contextos que se planteen. Por ello, el curso considera el desarrollo de laboratorios guiados e individuales, permitiendo la consolidación del aprendizaje del tema.

MATERIAL EDUCATIVO DE APRENDIZAJE

La relación de recursos de aprendizaje que se dispondrá son los siguientes:

Kit educativo del instructor		
Sílabo	Sí	Máquina virtual
Marco teórico	No	Vídeos recomendados
Diapositivas	Sí	Lecturas recomendadas
Análisis de caso de estudio	No	Tareas de refuerzo
Enunciado de laboratorio	Sí	Asesoría en línea
Guía de laboratorio	Si	Evaluaciones en línea
Solucionario de laboratorio	Sí	Kit de certificación
Repositorio en nube	Sí	Proyecto de investigación

Kit educativo del alumno		
Sílabo	Sí	Máquina virtual
Marco teórico	No	Vídeos recomendados
Diapositivas	Sí	Lecturas recomendadas
Análisis de caso de estudio	No	Tareas de refuerzo
Enunciado de laboratorio	Sí	Asesoría en línea
Guía de laboratorio	No	Evaluaciones en línea
Solucionario de laboratorio	Sí	Kit de certificación
Repositorio en nube	Sí	Proyecto de investigación

REQUERIMIENTOS PARA EL DICTADO DEL PROGRAMA

Recursos didácticos	Herramientas técnicas
<ul style="list-style-type: none"> • Teams • 1 máquina virtual para instructor • 1 máquina virtual por alumno • Conexión de red (física o inalámbrica) • Acceso directo en el escritorio de los alumnos a carpeta del instructor • Internet en la máquina del instructor y alumnos 	<ul style="list-style-type: none"> • Nodo Computacional <ul style="list-style-type: none"> – Procesador 4 vCPU – Memoria 16 GB – Disco 1 TB (minimo) • Software base <ul style="list-style-type: none"> ○ Ubuntu Server 18.04 LTS de 64 bits ○ Visual Code ○ MS Office 365 licencia • Servicios Cloud <ul style="list-style-type: none"> ○ Suscripción Azure Cibertec ○ Cuenta GitHub

COSTO DEL PROGRAMA

Descripción	Zona Urbana	Zona No Urbana	Provincia
Docente por hora	S/. 100.00		
Kit del alumno			
Taxi	--		

CONTENIDO TEMÁTICO

CAPÍTULO N° 1: Software Design Patterns and Principles	Duración 9h
--	----------------

Temas	Habilidades
<ul style="list-style-type: none"> • Paradigmas de programación • Principios SOLID (PART II, III) • Patron Singleton • Patron Factory Method • Patron Iterator • Patron Observer • Patron Proxy • Patron Bridge 	<p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Conocer los principios SOLID • Conocer los 6 patrones de diseño de software más importantes • Explorar código fuente con los patrones GoF

Laboratorios:

- Laboratorio C0101: Implementación del patron Singleton
- Laboratorio C0102: Implementación del patron Factory Method
- Laboratorio C0103: Implementación del patron Iterator
- Laboratorio C0104: Implementación del patron Observer
- Laboratorio C0105: Implementación del patron Proxy
- Laboratorio C0106: Implementación del patron Bridge

Tareas:

--

CAPÍTULO Nº 2: Design & Modeling of RESTful Web Services with a Study Use Case

Duración
6h

Temas

- Puntos claves para identificar los recursos de un servicio
- Overview REST
- Proceso de diseño de un servicio RESTful over HTTP

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Conocer los fundamentos del estilo arquitectónico REST
- Conocer los 10 pasos para construir un servicio REST sobre HTTP

Laboratorios:

- Laboratorio C0201: Diseñar un servicio HTTP/REST con un caso de estudio

Tareas:

--

CAPÍTULO Nº 3: Features of the framework for building RESTful Web Service

Duración
30 h

Temas

- DispatcherServlet
- Annotated Controllers
- Functional Endpoints
- URI Links
- HTTP Caching
- HTTP/2
- RestClients
- WebSockets

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Conocer a detalle los componentes necesarios del Spring Framework 5.x para la construcción de piezas de software con el protocolo HTTP/REST

Laboratorios:

- Laboratorio C0301: Diseñando el arquetipo para el caso de estudio
- Laboratorio C0302: Implementando los Controladores, URIs del servicio
- Laboratorio C0303: Implementando sockets para la obtención de la hora GMT

Tareas:

Implementar en el caso de estudio el feature de este capítulo

CAPÍTULO Nº 4: Just Run RESTful Service**Duración**
24 h**Temas**

- Using spring-boot
- Spring Application
- Externalized Configuration
- Profiles
- Logging
- JSON
- Calling RESTful w/ RestTemplate

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Facilitar la creación de aplicaciones independientes basadas en Spring que puede ser "simplemente ejecutada" y lista para su despliegue.

Laboratorios:

- Laboratorio C0401: Laboratorio de startup application
- Laboratorio C0402: Laboratorio de externalizacion de configuraciones
- Laboratorio C0403: Laboratorio de profiles dev, cer, pro
- Laboratorio C0404: Laboratorio de Logger
- Laboratorio C0405: Laboratorio de creacion de cliente REST Template

Tareas:

--

CAPÍTULO Nº 5: Accesing of Data from RESTful Service**Duración**
24 h**Temas**

- Trabajando con Spring Data Repository
- Spring Data JDBC
- Spring Data JPA
- Spring Data MongoDB
- Spring Data Redis

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Conocer los drivers sincronos de acceso a datos para el consumo de un backend SQL/NoSQL

Laboratorios:

- Laboratorio C0501: Laboratorio de JDBC
- Laboratorio C0502: Laboratorio de JPA
- Laboratorio C0503: Laboratorio de MongoDB
- Laboratorio C0504: Laboratorio de Redis

Tareas:

--

**CAPÍTULO N° 6: Features of the framework for building applications
integrating with messaging systems**

Duración
12 h

Temas

- Introducción AMQP
- AMQP vs JMS
- JMS Implementations
- amqp-template
- AMQP vendors
 1. Rabbit MQ Broker
 2. Azure Service Bus

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Conocer los fundamentos e implementar los componentes necesarios para la implementación del patrón de mensajería Publisher-Suscriber

Laboratorios:

- Laboratorio C0601: Laboratorio de creación de un servicio AMQP con RabbitMQ y Spring AMQP - Direct Exchange
- Laboratorio C0602: Laboratorio de creación de un servicio AMQP con Azure Service Bus y Spring Cloud Stream - Topic Exchange

Tareas:

--

CAPÍTULO N° 07: Reactive Programming

Duración
12 h

Temas

- Reactive Programming
- Reactive Stream
 - Manifest
 - Specification
 - Implementation
- Reactive Stream Implementations
- Operadores reactivos básicos
- Flowables y Backpressure

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Conocer los fundamentos iniciales de programación reactiva.
- Conocer las librerías reactivas.
- Entender la Programación reactiva con laboratorios.

Laboratorios:

- Laboratorio C0701: Laboratorio de Observables y Suscribers con Reactor
- Laboratorio C0702: Laboratorio de Operadores reactivos
- Laboratorio C0703: Laboratorio de backpressure

Tareas:

--

CAPÍTULO N° 08: Features of the framework for building non-blocking RESTful Service

Duración
15 h

- DispatcherHandler
- Annotated Controllers
- Functional Endpoints
- Reactive Core
- URI Links
- HTTP Caching
- HTTP/2
- WebSockets
- RSocket
- Reactive Libraries

Al finalizar el capítulo, el alumno podrá:

- Reconocer un proyecto WebFlux
- Implementar `@RestController` reactivos
- Diseñar un arquetipo para iniciar un proyecto reactivo

Laboratorios:

- Laboratorio C0801: Laboratorio de construcción de un servicio REST non-blocking

Tareas:

--

CAPÍTULO N° 09: Access to Reactive Stream Data Source

Duración
12 h

Temas

Habilidades

- Access Data Reactive R2
SQL Server, Mongo, Redis
- Spring Cloud Stream
Kafka, Event Hubs

Al finalizar el capítulo, el alumno podrá:

- Conocer los fundamentos del framework para la construcción de servicios

Laboratorios:

- Laboratorio C0901: Laboratorio de acceso a BD reactivo NoSQL
- Laboratorio C0902: Laboratorio de acceso a reactivo Cache Redis
- Laboratorio C0903: Laboratorio de creación de un servicio con API Event Hubs (Spring Cloud Stream)

Tareas:

--

CAPÍTULO Nº 10: Implementing security at backend service**Duración**
6 h**Temas****Architecture Components**

AuthenticationProvider

Authentication Mechanisms

OAuth 2.0 Login

OAuth 2.0 Client

OAuth 2.0 Resource Server

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Conocer los modelos de autentificacion OAuth y como se integran a nuestra pieza de software

Laboratorios:

- Laboratorio C1001: Laboratorio de construccion de un servicio REST non-blocking con tokens de autentificacion JWT

Tareas:

--

CAPÍTULO Nº 11: Protection of sensitive data with the vault**Duración**
6 h**Temas****Vault**

- Concepts
- Tecnologias
- Caso de Uso
- Azure Key Vault

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Integrar y proteger los datos sensibles de una pieza de software a un vault en cloud

Laboratorios:

- Laboratorio C1101: Laboratorio de construccion de un servicio REST non-blocking con Spring Cloud Key Vault Azure

Tareas:

--

CAPÍTULO Nº 12: Features of the framework for building SOAP Web Service**Duración**
3 h**Temas**

- What is Spring Web Service
- Why Contract First?
- Supports WS-Security
- Integrates with Spring Security

Habilidades

Al finalizar el capítulo, el alumno podrá:

- Implementar un servicio con el protocolo SOAP en Spring y las ventajas de el framework

Laboratorios:

- Laboratorio C1201: Laboratorio de construccion de un servicio SOAP con un backend en H2

Tareas:

