
ESPECIALIZACIÓN EN INGENIERÍA EN SISTEMAS DE INFORMACIÓN**MAESTRÍA EN INGENIERÍA EN SISTEMAS DE INFORMACIÓN**

1. Nombre de la actividad curricular: Desarrollo de Aplicaciones Orientadas a Servicios

2. Año Académico: 2020

3. Docente: Dr. Jorge Roa, Dr. Pablo David Villarreal

4. Fundamentación

En la actualidad, las organizaciones necesitan mantener competitividad en un mercado globalizado. Las organizaciones deben enfrentar condiciones de mercado cambiantes, nuevas presiones competitivas, leyes regulatorias que deben ser cumplidas, y amenazas de la competencia. Estas situaciones han llevado a la necesidad de que la Tecnología de Información (TI) deba responder rápido a los nuevos modelos de negocios y requerimientos.

Durante los últimos años, ha habido un creciente consenso en la industria de que la manera de crear soluciones de TI orientadas al negocio es el uso de bloques de construcción discretos de software, que se basan en protocolos estándares de la industria e interoperan a través de diferentes plataformas y lenguajes de programación. Esto dio origen a la Computación Orientada a Servicios (Service Oriented Computing, SOC).

SOC utiliza servicios como los constructores fundamentales para dar soporte al desarrollo rápido, de bajo costo y de fácil composición de aplicaciones distribuidas. Los servicios son autónomos, entidades independientes de la plataforma computacional que pueden ser utilizados de manera independiente de la plataforma. Pueden ser descriptos, publicados, descubiertos, y ensamblados dinámicamente para el desarrollo masivo de sistemas distribuidos, interoperables, y evolutivos. Los servicios ejecutan funciones tan variadas como responder a solicitudes simples o ejecutar procesos de negocio complejos que requieren relaciones punto-a-punto entre múltiples consumidores y proveedores de servicios.

Los Servicios Web es la tecnología más utilizada basada en el concepto de SOC. Proveen la base para el desarrollo y ejecución de procesos de negocio que están distribuidos a través de la red y disponibles mediante interfaces y protocolos estándares. Los Servicios Web utilizan estándares basados en Internet como medio de comunicación entre los servicios, y también para la definición y composición de los servicios.

Mediante la tecnología de Servicios Web, las organizaciones pueden exponer sus procesos de negocio principales como una colección de servicios en Internet. Los clientes y socios de negocio pueden invocar estos servicios a través de Internet y componerlos de manera tal que les permita alcanzar los objetivos de negocio. La aplicación de tecnologías de servicios permite reducir complejidad y costos, exponiendo y reusando las funcionalidades de negocio claves, incrementar la flexibilidad, adaptación en cambios de tecnologías, y mejoras en la eficiencia operativa.

La Arquitectura Orientada a Servicios (SOA, del inglés Service Oriented Architecture) es clave para estos conceptos. SOA es un estilo arquitectónico que propone una arquitectura por capas para diseñar un sistema de software que provee servicios a aplicaciones de usuarios finales o a servicios distribuidos en la red, mediante interfaces públicas. Los Servicios Web proveen una solución favorable para resolver problemas de integración entre sistemas de software autónomos y heterogéneos en una gran variedad de tecnologías.

El éxito de SOA radica en que permite la definición de sistemas de software con un bajo acoplamiento, lo que brinda flexibilidad y adaptabilidad. El bajo acoplamiento le facilita a un sistema la interacción con otros componentes de sistemas (incluso sistemas "legacy") que fueron desarrollados de manera independiente. Así, las aplicaciones orientadas a servicios pueden evolucionar más fácilmente

durante su tiempo de vida y adaptarse mejor a los cambios del entorno o incluso impredecibles.

En los últimos años, la necesidad de optimizar tiempos de entrega y despliegue de las aplicaciones y de optimizar el trabajo en equipos dio origen a la arquitectura de microservicios. Microservicios es una variante de SOA basada en descomponer una aplicación de software en diferentes servicios de menor tamaño para mejorar la modularidad y hacer que la aplicación sea más fácil de entender, desarrollar, probar y más resistente a los problemas.

Los servicios en Internet son cada vez más accesibles globalmente. Tanto en el sector público como en el privado se están adoptando servicios y aplicaciones orientadas a servicios para proveer funciones confiables y escalables. Mientras tanto, las tecnologías orientadas a servicios continúan evolucionando para dar soporte a las demandas de una mayor flexibilidad y respuestas ante ambientes dinámicos. Las organizaciones necesitan soluciones orientadas a servicios que consoliden sus procesos de negocio.

Por lo tanto, los profesionales en sistemas de información requieren conocer los conceptos teóricos y prácticos del dominio de la Computación y la Arquitectura Orientada a Servicios, y aplicarlos para llevar a cabo el desarrollo e implementación de aplicaciones orientadas a servicios. Para esto, el profesional en sistemas de información requiere adquirir conocimiento tanto en las metodologías y métodos para el diseño de aplicaciones orientadas a servicios, como así también en las tecnologías y lenguajes que permiten llevar a cabo la implementación de este tipo de aplicaciones.

5. Objetivos

El objetivo de este curso es introducir los conceptos teóricos y prácticos necesarios para el desarrollo de aplicaciones de software orientadas a servicios. Se pretende que el alumno conozca y sea capaz de aplicar diferentes métodos y lenguajes para llevar a cabo el diseño e implementación de aplicaciones de software basadas en los principios de la Computación Orientada a Servicios y de la Arquitectura Orientada a Servicios. Se pretende también que el alumno conozca, evalúe y aplique las tecnologías y estándares de Servicios Web disponibles para la implementación de este tipo de aplicaciones de software.

Al finalizar el curso, se pretende que el alumno adquiera los conocimientos en el tema y sea capaz de:

- diseñar la arquitectura de las aplicaciones orientadas a servicios, utilizando métodos para la identificación y definición de los servicios de acuerdo con los principios de SOC y SOA.
- utilizar y evaluar lenguajes y herramientas que den soporte al diseño e implementación de Servicios Web, tanto de arquitecturas basadas en servicios SOAP como aquellas basadas en servicios REST.
- utilizar y evaluar diferentes métodos y lenguajes para llevar a cabo la composición de Servicios Web.
- utilizar y evaluar diferentes métodos y lenguajes para llevar a cabo casos de prueba y la verificación de Servicios Web.
- comprender los conceptos de microservicios, su tipo de arquitectura y sus componentes principales.

6. Contenidos

Unidad 1: Aplicaciones y Arquitecturas Orientadas a Servicios

Aplicaciones basadas en Arquitecturas Orientadas a Servicios. Paradigma de Computación Orientada a Servicios. Concepto. Servicios simples y compuestos. Arquitectura Orientada a Servicios (SOA): conceptos y términos. SOA como arquitectura de sistemas distribuidos. SOA como arquitectura de organizaciones. SOA en la gestión de procesos de negocio. Ventajas y limitaciones de SOA.

Unidad 2: Diseño de aplicaciones orientadas a servicios

Principios de orientación a servicios. Métodos de diseño de servicios. Identificación y clasificación de servicios. Patrones de diseño de servicios.

Unidad 3: Servicios Web

Definición de Servicio Web. Tipos de Servicios Web: RESTful y el estándar SOAP. Aspectos de servicios web (implementación, acceso, contrato). Los lenguajes XML y JSON. Descripción de Servicios Web: el estándar WSDL (Web Service Description Language). Descubrimiento de Servicios Web: el estándar UDDI. Otros estándares de Servicios Web. Integración de Aplicaciones con Servicios Web.

Unidad 4: Desarrollo e implementación de Servicios Web.

Lenguajes y herramientas para la implementación de Servicios Web. Utilización de los frameworks JAX-WS, JAX-RS. Anotaciones. Parsers de XML: DOM y SAX. Mapeo documentos a objetos. Gestión de excepciones. Testeo y monitoreo de Servicios Web.

Unidad 5: Composición de Servicios Web

Conceptos de composición de Servicios Web. Orquestación de Servicios Web: el lenguaje WS-BPEL. Coreografía de Servicios Web: el lenguaje WS-CDL. El lenguaje BPMN para el modelado de orquestación y coreografía de Servicios Web.

Unidad 6: Verificación de Servicios Web

Concepto de verificación. Verificación de orquestación de Servicios Web. Verificación de coreografías de Servicios Web. Alineación de comportamiento entre servicios.

Unidad 7: Microservicios

Conceptos de microservicios. Principales beneficios. Aplicaciones monolíticas vs microservicios. Comunicación en una arquitectura de microservicios. Contenedores y microservicios. Principales desafíos.

7. Metodología de Enseñanza y Formación práctica

La mitad del tiempo del curso, las clases serán del tipo teórico-práctico, en donde se expondrán los conceptos teóricos y se realizará la resolución de guías de ejercicios con el propósito de afianzar los conocimientos.

La otra mitad del tiempo del curso será destinada a la formación práctica, a través de la resolución de guías con problemas y casos de estudio, como así también a la resolución de los trabajos prácticos, de tal manera que los alumnos puedan consultar al profesor los problemas o dudas que surjan de la elaboración de los mismos.

8. Carga horaria total

Carga horaria teórica	Carga horaria práctica	Carga horaria total
30	30	60

9. Modalidad de Evaluación

La evaluación de los conocimientos y capacidades adquiridas durante el cursado se realizará por medio de un examen escrito y la resolución de un trabajo práctico integral.

El examen escrito es integrador, final e individual.

El trabajo práctico se enfoca en la resolución de casos de estudios, utilizando los lenguajes, métodos y técnicas desarrollados en el cursado.

También se realiza un seguimiento de los alumnos durante el cursado, los cuales son expuestos al empleo de lenguajes, técnicas, métodos y herramientas, mediante la resolución de casos de estudio y guías de ejercicios prácticas, con el propósito de afianzar los conceptos teóricos.

10. Requisitos de aprobación y promoción

Para la aprobación del curso se requiere que los alumnos obtengan un 60% en cada instancia de evaluación. La nota final corresponderá en una ponderación de un 50% a la nota obtenida en el examen escrito y el otro 50% a la obtenida en el trabajo práctico.

11. Infraestructura y equipamiento

La infraestructura y ámbitos a utilizar en el dictado son los siguientes:

1. Campus virtual: el material bibliográfico del curso, las presentaciones y los enunciados de las ejercitaciones y trabajos prácticos se encuentran disponible en el campus virtual de la Facultad Regional Santa Fe.
2. Aulas: las clases teóricas se desarrollan en un aula con capacidad para 50 estudiantes, equipo de proyección y acceso a internet mediante conexión wifi. Todo el equipamiento mencionado es empleado en el dictado de las clases teóricas.
3. Laboratorio: se dispone de 2 laboratorios, LABSIS 4 y LABSIS 5 con capacidad para 20/25 estudiantes, en ambos casos con acceso a internet y disponibilidad de proyector. La utilización de uno u otro laboratorio depende de la cantidad de estudiantes inscriptos en el curso. En cuanto a software, en el desarrollo del curso se emplean herramientas y sistemas de diseño en implementación de servicios web open source (gratuitas), tales como Eclipse, Bonita, GlassFish Tools, Eclipse BPEL, SoapUI, JAX-WS, JAX-RS, Postman.

12. Bibliografía

1. Erl, Thomas. Service-oriented architecture: analysis and design for services and microservices. Prentice Hall, 2017.
2. Erl, Thomas. SOA Principles of Service Design. Prentice Hall, 2007.
3. Newman, Sam. Building microservices: designing fine-grained systems. " O'Reilly Media, Inc.", 2015.
4. Paik, Hye-young, et al. "Web Services–REST or Restful Services." Web Service Implementation and Composition Techniques. Springer, Cham, 2017. 67-91.
5. Zolotas, Christoforos, et al. "From requirements to source code: a Model-Driven Engineering approach for RESTful web services." Automated Software Engineering 24.4 (2017): 791-838.
6. Chen, Lianping. "Microservices: Architecting for Continuous Delivery and DevOps." IEEE International Conference on Software Architecture (ICSA). 2018.
7. Garriga, Martin, et al. "Web services composition mechanisms: a review." IETE Technical Review 32.5 (2015): 376-383.
8. Lemos, Angel Lagares, Florian Daniel, and Boualem Benatallah. "Web service composition: a survey of techniques and tools." ACM Computing Surveys (CSUR) 48.3 (2016): 33.
9. Daigneau, Robert. Service design patterns. Fundamental design solutions for SOAP/WSDL and RESTful Web Services. Addison Wesley 2012.
10. OASIS, Web Services Business Process Execution Language (2007, May), Available: <http://www.oasis-open.org/committees/download.php/23964/wsbpel-v2.0-primer.htm>.
11. Object Management Group (OMG). Business Process Modeling Notation Specification. Version 2.0. Object Management Group, 2010. www.bpmn.org
12. Roa, J., Villarreal, P., Chiotti, O. Behavior Alignment and Control Flow Verification of Process and Service Choreographies. Journal for Universal Computer Science, Special Issue on Conceptual Modelling of Services, Vol. 18, No. 17, pp. 2383-2406, ISSN: 0948-695x, 2012.

13. Villarreal, P., Salomone, H.E. and Chiotti, O.: Transforming Collaborative Business Process Models into Web Services Choreography Specifications. 2dn IEEE International Workshop on E-Commerce and Services, Lecture Notes in Computer Science, vol. 4055, pp 50-65, 2006.
14. Villarreal, Pablo. Método para el Modelado y Especificación de Procesos de Negocio Colaborativos. Tesis Doctoral. Universidad Tecnológica Nacional – Facultad Regional Santa Fe. ISBN 987-43-9708-8, paginas 357, 2005, CERIDE, Santa Fe, Argentina.
15. Villarreal, P, Lazarte, I., Roa, J., Chiotti, O. A Modeling Approach for Collaborative Business Processes based on the UP-ColBPIP Language. Business Process Management Workshops 2009. Lecture Notes in Business Information Processing (LNBIP), vol. 43, pp. 318-329, ISSN: 1865-1348, Springer-Verlag Berlin Heidelberg, 2010.
16. Roa, J., Chiotti, O., Villarreal, P. Specification of Behavioral Anti-Patterns for the Verification of Block-Structured Collaborative Business Processes. Information and Software Technology 75, 148 - 170 (2016).
17. Roa, Jorge. Métodos para la Verificación y Alineación de Comportamiento de Procesos de Negocio Colaborativos. Tesis Doctoral. Universidad Tecnológica Nacional – Facultad Regional Santa Fe. 2014, Santa Fe, Argentina.
18. Roa, J., Chiotti, O. Villarreal, P. A Verification Method for Collaborative Business Processes. En Business Process Management Workshops (BPMW), pp. 293-305, Springer Berlin Heidelberg, 2012.
19. Margolis, Ben, and Joseph Sharpe. SOA for the Business Developer: Concepts, BPEL, and SCA. Paris: Mc Press, 2007.
20. Erl, Thomas, Karmarkar, Anish, Priscilla Walmsley, Hugo Haas, L. Umit Yalcinalp, Kevin Liu, David Orchard, Andre Tost, and James Pasley. Web service contract design and versioning for SOA. Prentice Hall, 2009.