|  |  |  |
| --- | --- | --- |
|  | TÉCNICO SUPERIOR UNIVERSITARIO EN  TECNOLOGÍAS DE LA INFORMACIÓN ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA  EN COMPETENCIAS PROFESIONALES | descarga |

**ASIGNATURA DE APLICACIONES WEB ORIENTADAS A SERVICIOS**

|  |  |
| --- | --- |
| 1. **Competencias** | Implementar soluciones multiplataforma, en la nube y software embebido, en entornos seguros mediante la adquisición y administración de datos e ingeniería de software para contribuir a la automatización de los procesos en las organizaciones. |
| 1. **Cuatrimestre** | Cuarto |
| 1. **Horas Teóricas** | 32 |
| 1. **Horas Prácticas** | 73 |
| 1. **Horas Totales** | 105 |
| 1. **Horas Totales por Semana Cuatrimestre** | 7 |
| 1. **Objetivo de aprendizaje** | El alumno desarrollará aplicaciones Web híbridas orientadas a servicios mediante la integración de lenguajes de programación, frameworks de desarrollo y API's para la publicación en la nube. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Unidades de Aprendizaje** | **Horas** | | |
| **Teóricas** | **Prácticas** | **Totales** |
| I. Introducción al desarrollo Web orientado a servicios | 14 | 0 | 14 |
| II. Interfaz de Programación de Aplicaciones (API) | 10 | 39 | 49 |
| III. Integración de una aplicación orientada a servicios de tipo Mashup | 8 | 34 | 42 |
| **Totales** | **32** | **73** | **105** |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*UNIDADES DE APRENDIZAJE*

|  |  |
| --- | --- |
| 1. **Unidad de aprendizaje** | **I. Introducción al desarrollo Web orientado a servicios** |
| 1. **Horas Teóricas** | 14 |
| 1. **Horas Prácticas** | 0 |
| 1. **Horas Totales** | 14 |
| 1. **Objetivo de la Unidad de Aprendizaje** | El alumno identificará la arquitectura para el desarrollo de aplicaciones orientadas a servicios. |

| **Temas** | **Saber** | **Saber hacer** | **Ser** |
| --- | --- | --- | --- |
| Paradigma del desarrollo de aplicaciones orientadas a servicios | Distinguir los servicios que se ofrecen en la nube.  Identificar las características de las aplicaciones orientadas a servicios.  Identificar el concepto y las características de las aplicaciones Web hibridas (Mashup). |  | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Arquitectura orientada a servicios (SOA). | Definir la arquitectura orientada a servicios.  Identificar los principios de diseño que se aplican a sobre cada servicio modelado.  Identificar los estándares relacionados a los servicios: XML, SOAP, WSDL, UDDI, REST. | Identificar las ventajas de la arquitectura orientada a servicios.  Desarrollar diagramas de aplicaciones orientadas a servicios que integren los elementos de la arquitectura SOA. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |

**APLICACIONES WEB ORIENTADAS A SERVICIOS**

*PROCESO DE EVALUACIÓN*

|  |  |  |
| --- | --- | --- |
| **Resultado de aprendizaje** | **Secuencia de aprendizaje** | **Instrumentos y tipos de reactivos** |
| Desarrolla diagramas de aplicaciones orientadas a servicios bajo la arquitectura SOA que incluya las cuatro capas:   * La capa de acceso. * La capa de procesos. * La capa de servicio. * La capa de recursos. | 1. Identificar los servicios que se ofrecen en la nube que se pueden integrar en el desarrollo de aplicaciones.  2. Identificar las características de las aplicaciones Web orientadas a servicios.  3. Comprender la arquitectura orientada a servicios (SOA). | * Ejercicios prácticos. * Listas de cotejo. |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*PROCESO ENSEÑANZA APRENDIZAJE*

|  |  |
| --- | --- |
| **Métodos y técnicas de enseñanza** | **Medios y materiales didácticos** |
| * Discusión en grupo. * Mapas conceptuales. * Solución de problemas. | Pizarrón.  Plumones.  Computadora.  Internet.  Equipo multimedia.  Ejercicios prácticos.  Plataformas virtuales. |

*ESPACIO FORMATIVO*

|  |  |  |
| --- | --- | --- |
| **Aula** | **Laboratorio / Taller** | **Empresa** |
| **X** |  |  |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*UNIDADES DE APRENDIZAJE*

|  |  |
| --- | --- |
| 1. **Unidad de aprendizaje** | **II. Interfaz de Programación de Aplicaciones (API)** |
| 1. **Horas Teóricas** | 10 |
| 1. **Horas Prácticas** | 39 |
| 1. **Horas Totales** | 49 |
| 1. **Objetivo de la Unidad de Aprendizaje** | El alumno utilizará las API's para programar aplicaciones orientadas a servicios. |

| **Temas** | **Saber** | **Saber hacer** | **Ser** |
| --- | --- | --- | --- |
| Geolocalización. | Identificar las API's en el intercambio de información entre aplicaciones de geolocalización.  Identificar los elementos del lenguaje de programación necesarios para el uso de API's de geolocalización. | Programar aplicaciones con API's de geolocalización. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Redes sociales. | Identificar las API's en el intercambio de información entre aplicaciones de redes sociales.  Identificar los elementos del lenguaje de programación necesarios para el uso de API's de redes sociales. | Programar aplicaciones con API's de redes sociales. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| E-commerce. | Identificar las API's en el intercambio de información entre aplicaciones de e-commerce.  Identificar los elementos del lenguaje de programación necesarios para el uso de API's de e-commerce. | Programar aplicaciones con API's de e-commerce. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Bases de datos. | Identificar las API's en el intercambio de información entre aplicaciones de bases de datos.  Identificar los elementos del lenguaje de programación necesarios para el uso de API's de bases de datos. | Programar aplicaciones con API's de bases de datos. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Protocolos de comunicación. | Identificar las API's en el intercambio de información entre aplicaciones de protocolos de comunicación. | Programar aplicaciones con API's de protocolos de comunicación. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Plataformas on line. | Identificar las API's en el intercambio de información entre aplicaciones de plataformas on line.  Identificar los elementos del lenguaje de programación necesarios para el uso de API's de plataformas on line. | Programar aplicaciones con API's de plataformas on line. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Plataformas streaming. | Identificar las API's en el intercambio de información entre aplicaciones de plataformas streaming.  Identificar los elementos del lenguaje de programación necesarios para el uso de API's de streaming. | Programar aplicaciones con API's de plataformas streaming. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |

**APLICACIONES WEB ORIENTADAS A SERVICIOS**

*PROCESO DE EVALUACIÓN*

|  |  |  |
| --- | --- | --- |
| **Resultado de aprendizaje** | **Secuencia de aprendizaje** | **Instrumentos y tipos de reactivos** |
| Desarrolla un compendio de programas que incluya el uso de API's de:   * Geolocalización. * Redes sociales. * E-commerce. * Bases de datos. * Protocolos de comunicación. * Plataformas on line. * Plataformas streaming. | 1. Identificar las API's y sus funciones.  2. Comprender el funcionamiento de las API's.  3. Comprender el procedimiento de integración en una aplicación orientada a servicios. | * Ejercicios prácticos. * -Rúbrica. |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*PROCESO ENSEÑANZA APRENDIZAJE*

|  |  |
| --- | --- |
| **Métodos y técnicas de enseñanza** | **Medios y materiales didácticos** |
| * Práctica demostrativa. * Práctica en laboratorio. * Solución de problemas. | Pizarrón.  Plumones.  Computadora.  Internet.  Equipo multimedia.  Ejercicios prácticos.  Plataformas virtuales.  IDE de desarrollo. |

*ESPACIO FORMATIVO*

|  |  |  |
| --- | --- | --- |
| **Aula** | **Laboratorio / Taller** | **Empresa** |
|  | **X** |  |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*UNIDADES DE APRENDIZAJE*

|  |  |
| --- | --- |
| 1. **Unidad de aprendizaje** | **III. Integración de una aplicación orientada a servicios de tipo Mashup** |
| 1. **Horas Teóricas** | 8 |
| 1. **Horas Prácticas** | 34 |
| 1. **Horas Totales** | 42 |
| 1. **Objetivo de la Unidad de Aprendizaje** | El alumno desarrollará aplicaciones Web hibridas orientadas a servicios para su publicación en la nube. |

| **Temas** | **Saber** | **Saber hacer** | **Ser** |
| --- | --- | --- | --- |
| Diagrama de arquitectura SOAP para aplicaciones orientadas a servicios | Desarrollar el diagrama de aplicación Web hibridas (Mashup).  Seleccionar las API's a utilizar. |  | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Aplicaciones Web hibridas orientadas a servicios. | Identificar las plataformas y herramientas en el desarrollo de aplicaciones orientadas a servicios.  Identificar frameworks en el desarrollo de aplicaciones orientadas a servicios. | Programar aplicaciones que utilicen API's para conexión a servicios utilizando un framework de desarrollo. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |
| Publicación de aplicaciones Web hibridas orientadas a servicios. | Identificar el proceso de publicación de aplicaciones Web hibridas orientadas a servicios. | Diseñar el plan de validación de pruebas de funcionamiento.  Realizar la publicación de aplicaciones Web hibridas orientadas a servicios. | Analítico.  Lógico.  Ordenado.  Sistemático.  Creativo.  Propositivo. |

**APLICACIONES WEB ORIENTADAS A SERVICIOS**

*PROCESO DE EVALUACIÓN*

|  |  |  |
| --- | --- | --- |
| **Resultado de aprendizaje** | **Secuencia de aprendizaje** | **Instrumentos y tipos de reactivos** |
| Desarrolla aplicaciones Web híbridas que integren:   * Lenguajes de programación Web. * Frameworks para el desarrollo de aplicaciones Web. * API's incorporadas a las aplicaciones Web. | 1. Identificar las API's.  2. Identificar el lenguaje de programación y el Framework a utilizar.  3. Comprender la integración de elementos de las aplicaciones Web híbridas orientadas a servicios. | * Proyecto. * Rúbrica. |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*PROCESO ENSEÑANZA APRENDIZAJE*

|  |  |
| --- | --- |
| **Métodos y técnicas de enseñanza** | **Medios y materiales didácticos** |
| * Ejercicios prácticos. * Práctica demostrativa. * Aprendizaje basado en proyectos. | Pizarrón.  Plumones.  Computadora.  Internet.  Equipo multimedia.  Ejercicios prácticos.  Plataformas virtuales.  IDE de desarrollo. |

*ESPACIO FORMATIVO*

|  |  |  |
| --- | --- | --- |
| **Aula** | **Laboratorio / Taller** | **Empresa** |
|  | **X** |  |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA*

| **Capacidad** | **Criterios de Desempeño** |
| --- | --- |
| Identificar la propuesta de solución a través de técnicas y herramientas de modelado, para determinar los requerimientos técnicos del sistema de información. | Entrega un documento formal que incluya:   * Modelado de procesos: Casos de uso y diagrama de actividades * Recursos: Humanos, Materiales, Financieros y Tiempos * Riesgos * Partes involucradas * Propuesta de solución * Costo |
| Diseñar arquitectura del software mediante el modelado de los procesos y componentes para satisfacer los requerimientos técnicos y operacionales de la solución. | Entrega Un documento que incluya los diagramas UML de acuerdo a la propuesta de solución:   * Caso de uso * Clases * Secuencia * Actividades * Componentes * Colaboración * Estados * Distribución |
| Codificar soluciones de software seguras a través de entornos de desarrollo y arquitectura definida para su implementación. | Entrega el Código fuente documentado de la solución de software   * Métodos. * Atributos. * Variables. * Conexión a la base de datos. * Componentes. * Excepciones.   Pruebas unitarias:   * Diferentes escenarios de pruebas. * Criterios de aceptación. * Resultados de las pruebas. |
| Probar soluciones de software a través de ambientes automatizados de pruebas para garantizar que los resultados obtenidos sean los definidos en los requerimientos. | Entrega un Documento que incluya:   * Plan de pruebas * Criterios de aceptación * Resultados obtenidos de las pruebas * Aprobación de la solución |
| Implementar soluciones de software a través de la instalación y puesta en marcha para la liberación y cierre del proyecto. | Entrega la solución del software y lo documenta en:   1. Plan de instalación que incluya:  * Requerimientos de hardware y software * Requerimientos de infraestructura  1. Plan de puesta en marcha y operación  * Capacitación a usuarios * Pilotaje  1. Acta de cierre de proyecto:  * Empresa * Nombre del proyecto * Cliente * Líder del proyecto * Módulos * Fecha de entrega * Firma de aceptación |

# APLICACIONES WEB ORIENTADAS A SERVICIOS

*FUENTES BIBLIOGRÁFICAS*

| **Autor** | **Año** | **Título del Documento** | **Ciudad** | **País** | **Editorial** |
| --- | --- | --- | --- | --- | --- |
| Dennis Ashby and Claus T. Jensen | 2018/ 978111957138 | *API's for dummies. Tercera edición* | Hoboken | Estados Unidos | John Wiley & Sons, Inc. |
| Rupert Anderson | 2015/ 9781784394219 | *SoapUI Cookbook* | Birmingham | UK | Packt Publishing |
| Bill Burke | 2013 / 9781449361341 | *RESTful Java with JAX-RS 2.0* | Sebastopol | Estados Unidos | O'Reilly Media |
| Martin Kalin | 2013/ 9781449365110 | *Java Web Services: Up and Running* | Sebastopol | Estados Unidos | O'Reilly Media |
| Jobinesh Purushothaman | 2015/ 9781784399092 | *RESTful Java Web Services* | Birmingham | UK | Packt Publishing Limited |
| Thomas Erl, David Chou, John Devadoss, Nitin Gandhi, Hanu Kommalapati, Brian Loesgen, Christoph Schittko | 2017/ 9780134676357 | *Soa with .Net and Windows Azure: Realizing Service-Orientation with the Microsoft Platfor. Reimpresión 2017* | Boston | Estados Unidos | Prentice Hall |
| Thomas Erl | 2016/ 9780133858587 | *Service-Oriented Architecture: Analysis and Design for Services and Microservices* | Upper Saddle River | Estados Unidos | Prentice Hall |
| Thomas Erl | 2014/978-0133859034 | *SOA with Java: Realizing Service-Orientation with Java Technologies* | Upper Saddle River | Estados Unidos | Prentice Hall |
| Thomas Erl | 2016/ 978-0134524450 | *Service-Oriented Architecture (paperback): Concepts, Technology, and Design* | Upper Saddle River | Estados Unidos | Prentice Hall |
| Rhuan Rocha | 2018/ 9781788830621 | *Java EE 8 Design Patterns and Best Practices* | Birmingham | UK | Packt Publishing |
| Ganesan Senthilvel | 2017/9781786468888 | *Enterprise Application Architecture with .NET Core* | Birmingham | UK | Packt Publishing |
| Iuliana Cosmina | 2017/ 9781484228074 | *Pro Spring 5: An In-Depth Guide to the Spring Framework and Its Tools* | NY | Estados Unidos | Apress |
| Marten Deinum | 2017/ 9781484227893 | *Spring 5 Recipes: A Problem-Solution Approach* | NY | Estados Unidos | Apress |
| Dinesh Rajput | 2017/9781788299459 | *Spring 5 Design Patterns* | Birmingham | UK | Packt Publishing |
| Gaurav Aroraa | 2018/9781788291576 | *Building RESTful Web Services with .NET Core* | Birmingham | UK | Packt Publishing |
| Sanjay Patni | 2017/9781484226643 | *Pro RESTful APIs: Design, Build and Integrate with REST, JSON, XML and JAX-RS* | NY | Estados Unidos | Apress |
| Mario-Leander Reimer | 2018/9781789532883 | *Building RESTful Web Services with Java EE 8* | Birmingham | UK | Packt Publishing |
| Erich Gamma, Ralph Johnson, John Vlissides, Richard Helm, Grady Booch | 2015/ | *Design Patterns: Elements of Reusable Object-Oriented Software* | Chennai | India | Pearson |
| Christophe Aubry | 2017/ 978240900665-4 | *HTML5 y CSS3 Revolucione el diseño de sus sitios web (3ª edición)* | Barcelona | España | ENI |
| Steve Krug | 2014/ 9780321965516 | *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability* | NY | Estados Unidos | Pearson |
| [Mary Delamater; Ray Harris](https://www.bookfinder.com/search/?author=Mary%20Delamater&submit=Search&new_used=&currency=MXN&binding=&destination=mx&st=sr&ac=qr&lang=en) | 2015/ 9781890774851 | *Murach´s Javascript and DOM Scripting* | Birmingham | UK | Packt Publishing |