



1. Datos Generales de la asignatura

Nombre de la asignatura:	Tópicos de desarrollo de aplicaciones.
Clave de la asignatura:	TAD-1801
SATCA¹:	2-3-5 (Teóricas – Prácticas - Créditos)
Carrera:	<ul style="list-style-type: none"> ● Ingeniería en Sistemas Computacionales, ● Ingeniería en Tecnologías de la Información y Comunicaciones

2. Presentación

Caracterización de la asignatura

Esta asignatura, aporta al perfil del **Ingeniero en Sistemas Computacionales** e **Ingeniero en Tecnologías de la Información y Comunicaciones**, la capacidad de implementar métodos, herramientas y tecnologías contemporáneas en el desarrollo de aplicaciones interoperables para múltiples plataformas.

Su importancia radica, en que el estudiante adquiere los conocimientos necesarios para estar preparado a las nuevas exigencias de la industria de la creación de software. La asignatura, proporciona un panorama general de los roles que deben asumir los desarrolladores de software en la actualidad. Así mismo, nos muestra algunos de los ámbitos de alcance, en los que están directamente vinculados, para dar solución a problemas en múltiples plataformas.

Se ubica en el sexto semestre de la trayectoria escolar, por lo que se establece como una de las materias principales del módulo de especialidad, permitiendo que dé soporte a las otras materias del módulo y para el desarrollo de prácticas profesionales.

Intención didáctica

El enfoque sugerido para la materia, requiere que las actividades sean en principio, orientadas a la búsqueda de información para comprender los roles que deben asumir los desarrolladores en la actualidad; propiciando procesos intelectuales como inducción-deducción y análisis-síntesis; las actividades teóricas se han descrito como actividades previas.

Para el tratamiento práctico de los temas, se sugiere generar prácticas que promuevan el desarrollo de habilidades para la resolución de problemas en múltiples plataformas, tales como: almacenamiento de datos y manejo de grandes volúmenes de información, cómputo en la nube y

¹ Sistema de Asignación y Transferencia de Créditos Académicos

sistemas embebidos, donde el estudiante adquiera la capacidad de aplicar los conocimientos en la práctica. Promoviendo el trabajo en equipo y que el profesor asume el papel de guía al estudiante en la construcción de su conocimiento.

En la primera unidad, se definen los roles del desarrollador completo de software en todos los ámbitos de su competencia.

La segunda unidad, se dan a conocer al estudiante los nuevos paradigmas de los sistemas de almacenamiento de datos y su necesidad para resolver problemáticas con el manejo de grandes volúmenes de información.

La unidad tres, permite adentrarnos tanto en el cómputo en la nube, como en los servicios que se que se deben generar para atender diversas problemáticas requeridas en las empresas, en cuanto a la disponibilidad de su información y la generación de reportes.

Por último, la unidad cuatro, nos permite conocer las aplicaciones multiplataforma, como herramientas para el desarrollo de software.

3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Instituto Tecnológico de Villahermosa, Junio 28, 2017.	Academia de: <ul style="list-style-type: none"> ● Ingeniería en Sistemas Computacionales. ● Ingeniería en Tecnologías de la Información y Comunicaciones del Instituto Tecnológico de Villahermosa.	Reunión para la elaboración de firmas de la especialidad.

4. Competencia(s) a desarrollar

Competencia(s) específica(s) de la asignatura
Implementa métodos, herramientas y tecnologías emergentes en el desarrollo de aplicaciones interoperables para múltiples plataformas.

5. Competencias previas

<ul style="list-style-type: none"> ● Aplica algoritmos y lenguajes de programación para diseñar e implementar soluciones a problemáticas del entorno. ● Diseña y desarrolla programas para la solución de problemas computacionales utilizando el paradigma orientado a objetos. ● Analiza requerimientos y diseña bases de datos para generar soluciones al tratamiento de información basándose en modelos y estándares.



- Instala, configura y administra un gestor de base de datos para el manejo de la información de una organización, optimizando la infraestructura computacional existente.

6. Temario

No.	Temas	Subtemas
1	Roles del desarrollador completo.	1.1. Full Stack. 1.1.1 FrontEnd. 1.1.2 BackEnd. 1.1.3 DevOps. 1.1.4 Agile. 1.1.5 DevTools. 1.1.6 Design. 1.1.7 Analytics. 1.1.8 Database.
2	Sistemas de almacenamiento de datos.	2.1. Nuevos paradigmas. 2.1.1 Diferencias de nuevos paradigmas y bases de datos relacionales. 2.2 NoSQL. 2.2.1 Tipos de bases de datos NoSQL. 2.2.2 CRUD con bases de datos NoSQL. 2.3 Big Data. 2.3.1 Introducción al Big Data. 2.3.2 Unidades de medida en Big Data. 2.3.3 MapReduce.
3	Cómputo y servicios en la nube.	3.1 La nube. 3.1.1 Tipos de nube. 3.1.2 Servicios de la nube (AWS). 3.1.3 Virtualización. 3.1.4 Seguridad en la nube. 3.2 Programación en la nube. 3.2.1 Software para la nube.
4	Aplicaciones multiplataforma.	4.1 Apps multiplataforma. 4.1.1 Apps embebidas. 4.1.2 Apps móviles. 4.1.3 Apps web. 4.1.4 Apps para dispositivos. 4.1.5 Apps interactivas. 4.1.6 Internet de las cosas.

7. Actividades de aprendizaje de los temas

1. Roles del desarrollador completo.	
Competencias	Actividades de aprendizaje
<p>Específica(s):</p> <ul style="list-style-type: none"> ● Identifica y reconoce la importancia de los roles del desarrollador completo en la construcción de software. <p>Genéricas:</p> <ul style="list-style-type: none"> ● Capacidad de abstracción, análisis y síntesis. ● Comunicación oral y escrita. ● Habilidad para buscar, procesar y analizar información procedente de fuentes diversas. ● Capacidad de trabajo en equipo. ● Compromiso ético. ● Capacidad de aplicar los conocimientos en la práctica. ● Habilidad para trabajar en forma autónoma. 	<ul style="list-style-type: none"> ● Investigar los roles de un desarrollador completo de software. ● Escribir un reporte donde se establezca la importancia de los roles del desarrollador completo de software que necesitan las empresas en la actualidad. ● Hacer un mapa mental de la funciones de un desarrollador completo de software. ● Describir las funciones de FrontEnd. ● Describir las funciones de BackEnd. ● Describir las funciones de DevOps. ● Describir las funciones de Agile. ● Describir las funciones de DevTools. ● Describir las funciones de Design. ● Describir las funciones de Analytics. ● Describir las funciones de Database. ● Hacer un cuadro sinóptico de la funciones del desarrollador de software. ● Investigar las herramientas que utilizan los desarrolladores completos de software. ● Hacer uso de frameworks que permita el desarrollo de software desde los diferentes roles de desarrollador.
2. Sistemas de almacenamiento de datos.	
Competencias	Actividades de aprendizaje
<p>Específica(s):</p> <ul style="list-style-type: none"> ● Usa bases de datos con nuevos paradigmas de almacenamiento de 	<ul style="list-style-type: none"> ● Investigar los nuevos paradigmas de almacenamiento de datos. ● Escribir un reporte donde se establezca la importancia de los nuevos paradigmas



<p>datos que permitan el manejo de grandes volúmenes de información.</p> <p>Genéricas:</p> <ul style="list-style-type: none"> ● Capacidad de abstracción, análisis y síntesis. ● Comunicación oral y escrita. ● Habilidad para buscar, procesar y analizar información procedente de fuentes diversas. ● Capacidad de trabajo en equipo. ● Compromiso ético. ● Capacidad de aplicar los conocimientos en la práctica. ● Habilidad para trabajar en forma autónoma. 	<p>de almacenamiento de datos para dar solución a grandes volúmenes de información que necesitan las empresas en la actualidad.</p> <ul style="list-style-type: none"> ● Establecer la diferencia entre NoSQL y bases de datos relacionales. ● Identificar los tipos de bases de datos NoSQL. ● Realizar ejercicios de CRUD con bases de datos NoSQL. ● Investigar los conceptos del Big Data. ● Investigar las unidades de medida en Big Data. ● Describir la Arquitectura del MapReduce ● Instalar frameworks tales como Apache Hadoop. ● Realizar ejercicios de MapReduce. ● Realizar ejercicios de MapReduce en NoSQL.
---	---

3. Cómputo y servicios en la nube.

Competencias	Actividades de aprendizaje
<p>Específica(s):</p> <ul style="list-style-type: none"> ● Usa el cómputo de la nube para ofrecer servicios que aseguren la alta disponibilidad de las aplicaciones y los datos. <p>Genéricas:</p> <ul style="list-style-type: none"> ● Capacidad de abstracción, análisis y síntesis. ● Comunicación oral y escrita. ● Habilidad para buscar, procesar y analizar información procedente de fuentes diversas. ● Capacidad de trabajo en equipo. ● Compromiso ético. 	<ul style="list-style-type: none"> ● Investigar los conceptos de nube, tipos de nube, servicios en la nube, virtualización y seguridad en la nube. ● Realice un cuadro comparativo de los tipos de nube. ● Realice un cuadro sinóptico de tipos de servicios ofrecidos en la nube. ● Escribir un reporte donde se establezca las ventajas y desventajas de disponer de la nube para dar soluciones a las empresas en la actualidad. ● Escriba un reporte sobre los aspectos de seguridad y las limitaciones en la nube. ● Investigue la herramientas para el desarrollo en la nube. ● Describa las ventajas y desventajas de



<ul style="list-style-type: none"> ● Capacidad de aplicar los conocimientos en la práctica. ● Habilidad para trabajar en forma autónoma. 	<p>las herramientas para el desarrollo en la nube.</p> <ul style="list-style-type: none"> ● Instale herramientas para el desarrollo en la nube. ● Utilice patrones de diseño para el desarrollo en la nube. (Retry Pattern, Valet Key y Sharding Pattern) ● Pruebe IDE para el desarrollo en la nube.
4. Aplicaciones multiplataforma.	
Competencias	Actividades de aprendizaje
<p>Específica(s):</p> <ul style="list-style-type: none"> ● Crea aplicaciones multiplataforma y para el internet de las cosas. <p>Genéricas:</p> <ul style="list-style-type: none"> ● Capacidad de abstracción, análisis y síntesis. ● Comunicación oral y escrita. ● Habilidad para buscar, procesar y analizar información procedente de fuentes diversas. ● Capacidad de trabajo en equipo. ● Compromiso ético. ● Capacidad de aplicar los conocimientos en la práctica. ● Habilidad para trabajar en forma autónoma. 	<ul style="list-style-type: none"> ● Investigar los conceptos de plataforma, tipos de plataforma y aplicaciones multiplataforma. ● Clasificar las diversas aplicaciones multiplataforma. ● Haga un reporte sobre el internet de las cosas y su importancia en la actualidad. ● Pruebe IDE para el desarrollo de aplicaciones multiplataforma. ● Cree aplicaciones multiplataforma. ● Realice ejercicios con Apps embebidas. ● Realice ejercicios con Apps móviles. ● Realice ejercicios con Apps web. ● Realice ejercicios con Apps para dispositivos. ● Realice ejercicios con Apps interactivas. ● Realice ejercicios con Apps para el Internet de las cosas.

8. Práctica(s)

<ul style="list-style-type: none"> ● Instalar frameworks que permita el desarrollo de software desde los diferentes roles de desarrollador. ● Construya una aplicación web backend. ● Construya una aplicación web frontend. ● Construya una aplicación web que cumpla backend y frontend al mismo tiempo. ● Editar, compilar y ejecutar distintos programas que incluyan el uso de formularios,



botones, etiquetas y cajas de texto.

- Desarrollar aplicaciones que incluyan la programación de eventos.
- Desarrollar aplicaciones que incluyan la generación de nuevos eventos.
- Desarrollar aplicaciones que maneje gráficos en un componente tipo canvas.
- Desarrollar aplicaciones que maneje componentes visuales y no visuales proporcionados por la IDE.
- Desarrollar aplicaciones que maneje librerías proporcionados por la IDE.
- Desarrollar componentes visuales o no visuales a partir de requerimientos previamente definidos y aplicarlos en el diseño de aplicaciones.
- Desarrollar un juego para un dispositivo móvil.
- Diseñar la base de datos en el modelo E-R del planteamiento de un problema real, convertir el esquema conceptual usando NoSQL .
- Instale y use servidores de bases de datos NoSQL.
- Realizar ejercicios de CRUD con bases de datos NoSQL.
- Instalar frameworks tales como Apache Hadoop.
- Realizar ejercicios de MapReduce.
- Realizar ejercicios de MapReduce en NoSQL.
- Utilice patrones de diseño para el desarrollo en la nube. (Retry Pattern, Valet Key y Sharding Pattern)
- Pruebe IDE para el desarrollo en la nube.
- Realice ejercicios con Apps embebidas.
- Realice ejercicios con Apps móviles.
- Realice ejercicios con Apps web.
- Realice ejercicios con Apps para dispositivos.
- Realice ejercicios con Apps interactivas.
- Realice ejercicios con Apps para el Internet de las cosas.

9. Proyecto de asignatura

El objetivo del proyecto que plantee el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:

- **Fundamentación:** marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo.
- **Planeación:** con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso:



de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.

- **Ejecución:** consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.
- **Evaluación:** es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

10. Evaluación por competencias

La evaluación debe ser permanente y continua. Se debe hacer una evaluación diagnóstica, formativa y sumativa. Se debe aplicar autoevaluación, coevaluación y heteroevaluación.

Algunos de los instrumentos que se pueden utilizar:

- Mapa conceptual.
- Mapa mental.
- Guía de problemas de programación.
- Examen (teóricos y prácticos).
- Reportes de prácticas.
- Resúmenes.
- Cuadro sinóptico.
- Preguntas guiadas.
- Plenaria.
- Cuadro comparativo.

Herramientas:

- Rúbrica.
- Lista de cotejo.
- Matriz de valoración.
- Guía de observación.

Se debe generar un portafolio de evidencias, de preferencia en formato digital.

11. Fuentes de información

- Colin J Ihrig y Adam Bretz, Full Stack JavaScript Development With MEAN: MongoDB, Express, AngularJS y NodeJS, SitePoint, 2014.



- Kathleen Taylor y Bud E. Smith, Getting a Web Development Job For Dummies, John Wiley & Sons, Inc., 2015.
- Azat Mardan, Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB, Apress, 2015.
- Fabian Vogelsteller, Isaac Strack y Marcelo Reyna, Meteor: Full-Stack Web Application Development, Packt Publishing, 2016.
- Patrick Mulder, Full Stack Web Development with Backbone.js: Scalable Application Design with 100% Javascript, O'Reilly, 2014.
- Adam Freeman, PRO AngularJS, Apress, 2014.
- Alex Young y Marc Harter, NodeJS in Practice, Manning Publications Co, 2014.
- Basarat Ali Syed, Beginning NodeJS, Apress, 2014.
- Bear Bibeault y Yehuda Katz, "jQuery in action", 2010, Manning Publications Co, 2010.
- Brett McLaughlin, What is Node, O'Reilly, 2014.
- Javier Eguíluz Pérez, "Introducción a Javascript", licencia Creative Commons, www.librosweb.es, 2009.
- Jonathan Chaffer y Karl Swedberg, "Learning JQuery, Third Edition", Packt Publishing, 2011.
- Ken Williamson, Learning AngularJS, O' Reilly, 2015.
- Matt Frisbie, AngularJS Web Application Development Cookbook, Packt Publishing, 2014.
- Maximiliano Firtman, "Ajax con JQuery", 2011, Ed. Alfaomega.
- Mike Cantelon, Marc Harter, TJ Holowaychuck, Nathan Rajlich, NodeJS in action, Manning Publications Co, 2014.
- Ángel Arias, Computación en la Nube: 2ª Edición, IT Campus Academy, 2015.
- Juan Ernesto Chávez Pacheco, Salvador Álvarez B., Chadwick Carreto A., Servicio, Disponibilidad e Interconexión para el Cómputo en la Nube: Modelo y Arquitectura para la Gestión del Servicio de Disponibilidad e Interconexión para el Cómputo en la Nube, Editorial Académica Española, 2013.
- Contreras Castañeda, Miguel Ángel, Desarrollo de aplicaciones Web multiplataforma, Aula Mentor, 2016.
- Francisco Javier Ceballos Sierra, Aplicaciones .NET multiplataforma, RA-MA S.A. Editorial y Publicaciones, 2008.

Sitios web de referencia y consulta:

- <http://jquery.com/>
- <http://jqueryui.com/>
- <http://librojquery.com/>
- <https://angularjs.org/>
- <http://www.w3schools.com/angular/>
- <http://www.codecademy.com/es/learn/learn-angularjs>
- <http://www.desarrolloweb.com/manuales/manual-angularjs.html>
- <http://angularjs.blogspot.com.ar/>
- <https://nodejs.org/>
- <http://nodeschool.io/es/>



- <http://www.desarrolloweb.com/manuales/manual-nodejs.html>
- <https://www.quora.com/What-is-a-Full-Stack-Web-framework>
- <https://bbvaopen4u.com/es/actualidad/que-es-un-full-stack-developer>