

# Enseñanza de Tópicos en Programación Web usando Resultados de Aprendizaje Basados en Evidencias

## Teaching Topics in Web Programming using Evidence-Based Learning Results

Juan Carlos López Pimentel , Jesús Carlos Sánchez Guzmán, Alicia González Laguna, Imelda Valles López, Octavio Ariosto Ríos Tercero, María Candelaria Gutiérrez Gómez , Francisco Suárez Ruíz

Tecnológico Nacional de México - Instituto Tecnológico de Tuxtla Gutiérrez  
Departamento de Sistemas y Computación  
Carretera Panamericana Km. 1080, Tuxtla Gutiérrez, Chiapas, C.P. 29050  
jclopezpimentel@gmail.com, jesuscarlos@ittg.edu.mx, agonzalezl@ittg.edu.mx,  
imevalles@yahoo.com.mx, orioست@ittg.edu.mx, mgutierrez@ittg.edu.mx,  
chontaes@hotmail.com

### PALABRAS CLAVE:

Programación, Web, Cliente-Servidor, Estrategias de Enseñanza y Aprendizaje.

### RESUMEN

Este artículo presenta nuestra experiencia y conocimiento en la enseñanza de desarrollo de sitios web. El objetivo de este artículo es mostrar un método de enseñanza en un curso llamado Programación Web. El curso lo hemos impartido en dos escenarios distintos; uno es localizado en el último semestre de una materia de Ingeniería en Sistemas Computacionales. El otro en una materia intermedia dentro de Ingeniería en Desarrollo de Software. La materia está compuesta de 75 horas presenciales; sin embargo, para combatir la restricción de tiempo considerando la cantidad de tecnologías web que existen actualmente y dar a los estudiantes la principal motivación de aprender el Estado del Arte en este tópico, hemos seleccionado cuidadosamente un conjunto de tópicos importantes y representativos que eventualmente conducen a un conjunto de herramientas usadas por desarrolladores actuales. Además se muestran algunas actividades convertidas a evidencias que los estudiantes deben realizar para completar el contenido del curso y aprobarlo. Este artículo también puede ser usado por otros profesores quienes están buscando una estrategia de enseñanza en cursos avanzados de Tecnologías Web.

### KEYWORDS:

Programming, Web, Client-Server, Teaching Strategy and Learning.

### ABSTRACT

This article presents our experience and knowledge in teaching website development. The aim of this article is to show a teaching method in a course so-called Web Programming. The course has been given in two different scenarios; one is located in the last semester in Computer Systems Engineering; the other in an intermediate period in Software Development Engineering. The course is composed of 75 classroom hours; However, in order to combat the restriction of time considering the number of web technologies that currently there exist and give students the main motivation to learn the State of Art in this topic, we have selected a set of important and representative topics that eventually led to a set of tools used by current developers. In addition there are some activities that become evidences that the students must develop to complete the content of the course and to approve it. Other teachers who are looking for a teaching strategy in advanced courses of Web Technologies can also use this article.

Recibido: 4 de agosto del 2017 • Aceptado: 3 de marzo del 2018 • Publicado en línea: 28 de agosto del 2018

## 1. INTRODUCCIÓN

La Programación Web es una forma de diseñar y desarrollar aplicaciones que se utilizarán en ambientes web; estas aplicaciones proporcionan al usuario la posibilidad de utilizarlas sin importar su ubicación ya que la mayoría de las computadoras poseen de forma predeterminada un navegador web y acceso a Internet.

Actualmente las aplicaciones en ambiente Web son tan populares como aquellas desarrolladas para dispositivos móviles. Inclusive hay ambientes de desarrollo Web para dispositivos móviles. De ahí la gran importancia de los diferentes temas que ameritan verse en tópicos que tienen que ver con el Desarrollo de Aplicaciones basadas en Web.

Esta nueva era con el Internet demanda y desafía a los docentes enfocados a Tecnologías de Información (TI) a reformar cursos tradicionales de desarrollo Web. Actualmente, existe un amplio rango de temas, y es difícil encontrar un simple libro de texto que sea apropiado para el curso. Todo esto sin contar que muchas soluciones de programación pueden encontrarse en la red, siendo esto un verdadero reto para los docentes, el diseñar prácticas reto extra-clase para sus estudiantes.

Existen diversos autores que han propuesto diferentes estrategias de cómo afrontar la enseñanza de la Programación Web, [9, 7, 4, 5], de las cuáles se han retomado algunos aspectos que citaremos más adelante.

La materia de Programación Web la hemos impartido en dos escenarios distintos, una llamada Tópicos Avanzados en Tecnologías Web es una materia que actualmente está ubicada en el último semestre de la carrera de Ingeniería en Sistemas Computacionales, eso significa que los estudiantes deben estar altamente capacitados en temas como programación estructurada, programación orientada a objetos, estructura de datos, base de datos e inclusive haber llevado materias de programación visual (ya sea en ambiente escritorio o móvil). El otro escenario ha sido en un curso intermedio de la carrera de Ingeniería en Desarrollo de Software; sin embargo, en este último escenario los alumnos aún no han llevado la materia de Base de Datos.

Un problema adicional es, que aunque, hay una cantidad abundante de tutoriales en programación web, no es fácil para los estudiantes con poca experiencia validar la relevancia de los recursos disponibles en la web, es decir, algunos estudiantes necesitan ser guiados y enseñados.

Así pues, estamos inspirados en la siguiente

hipótesis: Utilizando una estrategia donde la experiencia del profesor y la contribución de los mismos estudiantes en la generación de materiales dinámicos para el aprendizaje, podemos encontrar la fórmula para lograr las competencias necesarias que deben adquirir los estudiantes en la Programación Web en tiempos tan demandantes.

El objetivo general de este artículo es proporcionar a facilitadores, e inclusive alumnos, una metodología de enseñanza-aprendizaje para la materia de Tópicos en Programación Web a través de estrategias que combinan el conocimiento y la experiencia del profesor junto a la contribución de los mismos estudiantes en la generación de material dinámico para el aprendizaje; todo esto eventualmente conducirá a tener mejores competencias en los estudiantes.

La principal contribución de este artículo es proveer una alternativa de enseñanza-aprendizaje en la Programación Web. Esta alternativa es adecuada siempre y cuando el profesor tenga enfrente a estudiantes de grados avanzados. Otra contribución, pero no menos importante, es proveer una serie de temas que pueden ser vistos en una materia que tiene que ver con la Programación Web. Esta estrategia se ha aplicado en las carreras de Ingeniería en Desarrollo de Software e Ingeniería en Sistemas Computacionales perteneciente a las Universidades Politécnicas y a los Tecnológicos Nacionales respectivamente; para ambos sistemas se dan recomendaciones en la sección 8.

El resto de este documento se organiza de la siguiente manera: primeramente damos un pequeño resumen de algunos trabajos que se relacionan a este; posteriormente especificamos las unidades de aprendizaje que se han trabajado en la materia de Programación Web; después se especifican las evidencias que los estudiantes deberán presentar para demostrar sus conocimientos y habilidades; las secciones 5-7 muestran a detalle nuestro método de enseñanza y explicación a detalle de las actividades. Posteriormente se muestran algunos resultados, rendimientos y reflexiones obtenidas. Finalmente damos algunas conclusiones sobre el trabajo realizado.

## 2. TRABAJOS RELACIONADOS

Wang and Zahadat, [9], han considerado que el crecimiento explosivo de Tecnologías Web 2.0 presentan un desafío significativo para los profesores enfocados a enseñar desarrollo Web. Stelios Xinogalos and Theodores

H, [7], han desarrollado un estudio importante sobre los principales desafíos en la enseñanza de la programación Web. Yi Liu, [4], presenta un estudio similar pero también incluye herramientas importantes a usar en la enseñanza de la Programación Web.

Wang and Zahadat, [9], han propuesto un método que consiste en 4 elementos: a) Concentrase en AJAX; b) Dividir el curso en dos: uno enfocado al cliente y otro al servidor; c) Asignar proyectos que integren tópicos con aplicaciones del mundo real; y d) Usar métodos de enseñanza constructivista.

Francesco Maiorana, [5], discute un caso de estudio basado sobre la administración de un sistema de logueo, cubriendo con eso principales tópicos de programación web, bases de datos (transacciones, procedimientos almacenados) y aspectos de seguridad. Nosotros tomamos en cuenta parte de dicho caso de estudio en una de nuestras prácticas propuestas (ver tercera práctica en sección 6.2), sin embargo, no es la parte central y no tomamos en cuenta por lo pronto aspectos de seguridad en profundidad.

A diferencia de Wang y Zahadat, [9], (mencionado anteriormente) nuestro principal desafío es ajustar todos los tópicos en un solo curso que contiene 80 horas presenciales y 40 horas extra clase (120 en total). Sabemos que no es fácil para los estudiantes con poca experiencia en programación web absorber tanta información en un período de tiempo corto. A continuación detallamos un poco más sobre las unidades.

### 3. UNIDAD DE APRENDIZAJE

La materia comprende 15 semanas, cada semana involucra 5 horas presenciales y 2 horas extra-clase, las horas presenciales se dividen en 2 horas teoría y 3 horas prácticas. Las unidades trabajadas en la materia son:

- Diseño, desarrollo y programación de páginas web de lado del cliente.
- Internet y su arquitectura
- Programación de páginas web de lado del servidor.
- Comunicación asíncrona
- Desarrollo de sitios web

#### 3.1. DISEÑO, DESARROLLO Y PROGRAMACIÓN DE PÁGINAS WEB DE LADO DEL CLIENTE.

Esta unidad comprende tres resultados de aprendizaje:

- Diferenciar entre las diferentes versiones de HTML y CSS.
- Diseñar páginas web del lado del cliente usando tecnologías HTML5 Y CSS3.
- Desarrollar programas que involucren el uso del lenguaje de programación del lado del cliente (JavaScript) y relacionarlo con HTML y CSS.

#### 3.2. INTERNET Y SU ARQUITECTURA

La segunda unidad comprende tres resultados de aprendizaje:

- Describir la perspectiva histórica de Internet, la familia de protocolos TCP/IP.
- Conocer la arquitectura Cliente/Servidor enfocándose y detallando el funcionamiento del protocolo HTTP.
- Instalar servidores web y base de datos para desarrollar sitios web.

#### 3.3. PROGRAMACIÓN DE PÁGINAS WEB DE LADO DEL SERVIDOR

La tercera unidad comprende tres resultados de aprendizaje:

- Identificar las tecnologías existentes para programación web del lado del servidor.
- Conocer el paradigma de programación web del lado del servidor (variables de sesión, cookies, etc.).
- Desarrollar programas utilizando un lenguaje del lado del servidor.

#### 3.4. COMUNICACIÓN ASÍNCRONA

La cuarta unidad comprende tres resultados de aprendizaje:

- Identificar la diferencia entre comunicación síncrona y asíncrona.
- Conocer Tecnologías que manejan eventos de tiempo en el lado del cliente, AJAX, y Web Workers.
- Desarrollar programas utilizando tecnologías AJAX y Web Workers.

#### 3.5. DESARROLLO DE SITIOS WEB

La quinta unidad comprende tres resultados de aprendizaje:

- Conocer el uso de frameworks para el desarrollo

- de aplicaciones web y móviles.
- Utilizar las funciones para acceder a base de datos desde un servidor web para aplicar las Tecnologías síncronas y asíncronas en aplicaciones web.
- Desarrollar programas del lado del cliente y servidor con acceso a base de datos en ambiente web.

#### 4. METODOLOGÍA BASADA EN EVIDENCIAS DE APRENDIZAJE

La enseñanza de cursos de programación en general son difíciles, y los de programación web no son la excepción. Decidir qué tipo de actividades y proyectos se utilizará en un curso de estos, así como lograr un buen equilibrio entre la parte teórica y la práctica, es bastante difícil. Aquí proponemos nueve evidencias ; Clasificadas en: a) Reglas iniciales del curso; b) Conocimiento; c) Prácticas; y d) Proyecto. Cada una de las evidencias se le ha asignado un porcentaje específico con respecto a la puntuación general del curso (Ver Tabla 1).

Tabla 1: Evidencias generales de un curso en Programación Web

Tipo	Evidencia	%
Reglas iniciales del curso (saber ser)	Compromiso	5 %
Conocimiento (Saber)	Presentación Oral	10 %
	Diseño de Práctica	10 %
	Examen Final	15 %
Prácticas (Saber hacer)	Lado del Cliente	10 %
	Lado del Servidor	10 %
	Interacción Cliente-Servidor	10 %
Proyecto (saber hacer)	Especificación inicial del proyecto	10 %
	Especificación Final	20 %

Los estudiantes deben obligatoriamente obtener por lo menos (70/100) como promedio general de todas las evidencias para aprobar el curso. A lo largo de la impartición de este método de enseñanza-aprendizaje, se ha identificado la dificultad por parte de los estudiantes en alcanzar el 100 % de la calificación final. Para combatir esa restricción y tomando en cuenta la inspiración de los estudiantes se ha promovido dos estrategias para que los alumnos puedan obtener una calificación extra. Por un lado las prácticas retos y por otro lado, la participación, asistencia y participación en

actividades extracurriculares. La Figura 1 muestra un resumen del método propuesto.

#### 4.1. REGLAS INICIALES DEL CURSO (SABER SER)

La evidencia Políticas del Curso ayuda a que los estudiantes sepan las principales reglas y políticas del curso. Es importante desmenuzarlas en políticas dentro del salón de clases, sobre las actividades, asistencia, etc. Si el estudiante sabe cómo le gusta trabajar al docente y el alumno las cumple, tiene altas probabilidades de aprobar el curso. En esta etapa es importante que el estudiante sepa el Learning Management System que usará el profesor durante la administración general del curso (en algunas escuelas esto ya es proveído como una política general).

Se sugiere en esta actividad que el estudiante firme las políticas del curso y escriba a qué se compromete con el curso, esto puede implicar una reflexión con el estudiante al comienzo y al final del curso.

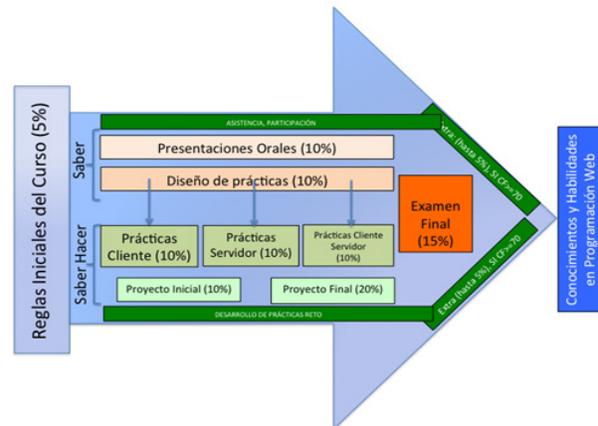


Figura 1: Descripción gráfica de nuestro método de enseñanza en Programación Web

#### 4.2. CONOCIMIENTO (SABER)

El desarrollo de aplicaciones web demanda la combinación de varias tecnologías. Además, las opciones de desarrollo tanto del lado del cliente como del servidor son muy numerosas. Seleccionar los tópicos y las tecnologías web fundamentales para que se cubran en un curso de programación web, además de proveer la profundidad apropiada por cada tópico es uno de los primeros desafíos que tienen los instructores. Si tomamos en cuenta la disponibilidad de Tecnologías de Información actuales, estudiantes

podrían en un momento dado profundizar en algunos tópicos y encontrar aspectos y temas novedosos que quizá el profesor sólo no podría abarcarlo. De ahí que se propone una actividad de Exposición, seleccionando algunos temas que los estudiantes deberían prepararlo y presentarlo. Para complementar esta actividad los estudiantes deben diseñar una práctica que debe ser resuelta por sus propios compañeros. En esta actividad los estudiantes toman el rol de facilitadores (al menos una vez durante todo el curso). Adicionalmente, se les aplica una pequeña prueba para demostrar que han adquirido por lo menos los conocimientos básicos. Más detalles en la sección 5.

#### 4.3. PRÁCTICAS

Se ha desarrollado tres conjuntos de diferentes prácticas, relacionados a los tópicos vistos en las secciones 3.1, 3.2, 3.3, 3.4 y 3.5. Las prácticas pueden ser obtenidas de libros, recursos en línea, diseñados por el mismo instructor y por los mismos estudiantes (para esta última ver la sección 4.2 para más detalles). Es usual que los estudiantes a través de diferentes generaciones compartan sus soluciones de sus trabajos. Para evitar esto, se sugiere que un porcentaje de la actividad sea diseñada por los mismos estudiantes en cada curso (ver sección 4.2). Con esto podemos evitar tener las mismas prácticas todo el tiempo.

#### 4.4. PROYECTO

Pensar en un proyecto final donde los estudiantes puedan empezar y explayarse en cuanto a diseño y desarrollo no es una tarea sencilla. Sin embargo, aquí proponemos un proyecto que nos ha dado muy buenos resultados y que cada vez hemos pulido más durante los últimos 7 años. Nuestra propuesta consiste que cada estudiante replique el diseño y desarrolle un Simulador de Sistema Operativo. Esto involucra para el estudiante elegir un Sistema Operativo existente para replicarlo en cuanto a diseño y desarrollar algunos módulos para que realmente se llegue a una simulación. Esta propuesta está dividida en dos partes: diseño y desarrollo. La parte de diseño involucra temas como HTML, XML, CSS y JavaScript. La parte de desarrollo involucra el proyecto final y se deja cuando ya se han visto temas de programación del lado del servidor. En la sección 7 describimos con mayor detalle esta actividad.

## 5. EVALUACIÓN DE CONOCIMIENTO

### 5.1. PRESENTACIÓN ORAL

#### 5.1.1. TÓPICOS

Seleccionar los temas y las tecnologías web subyacentes cubiertas en un curso de programación web, así como proporcionar la profundidad adecuada para cada tema, es el primer desafío que enfrenta un instructor. Si tenemos en cuenta la disponibilidad actual de tecnología de la información, los estudiantes podrían en un momento dado profundizar en algunos temas y encontrar temas nuevos e interesantes que tal vez el profesor no podría cubrirlos si los hace solo. Es por ello que se propone una actividad expositiva, seleccionando algunos temas que los alumnos deben preparar para presentarlo. Esta actividad debe ser realizada en diferentes equipos, sería formidable poder distribuir los temas en equipos que tengan cierta experiencia en alguno de los temas, aunque no es necesario porque se recomienda que el profesor siempre imparta el tema introductorio antes que los estudiantes les toque su participación. Los temas son los siguientes:

- HTML vs HTML5
- CSS vs CSS3
- JavaScript
- HTML DOM y JavaScript
- HTML Avanzado (Graphics, Canvas, SVG, Icons, Google Maps, Games)
- PHP primera parte
- JQuery
- PHP segunda parte
- Tecnologías para diseño en ambiente móvil: JQueryMobile
- Tecnologías para diseño en ambiente móvil: Phoneygap

#### 5.1.2. REQUERIMIENTOS

Sobre cada tema se recomienda a los estudiantes sigan los siguientes aspectos:

- Realizar una investigación enfocada a conocer en detalle la estructura sintáctica y semántica para el desarrollo de aplicaciones del subtema otorgado. Esta parte no debería dificultarse a los estudiantes ya que deben tener un buen background considerando que es una materia que se imparte en los últimos semestres.

- Diseñar una presentación usando html y la tecnología usada del subtema que les fue otorgado. El objetivo es que los estudiantes a medida que investigan del tema que les tocó también vayan fortaleciendo sus habilidades practicando con el diseño de su presentación.
- Mostrar y desarrollar ejemplos y explicarlos. Es importante que los estudiantes no solo muestren los ejemplos en clase, sino que también lo vayan elaborando, esto les permite comprobar lo que explican y reafirmar su conocimiento.
- Programar la presentación para que tenga una duración máxima de 50 minutos. Proporcionamos una posible lista de cotejo, la cual puede servir de guía. Puede descargarse de <https://checkListExpo.pdf>

## 5.2. DISEÑO DE PRÁCTICA

Para complementar la actividad descrita en la sección 5.1, los estudiantes deben diseñar una práctica que deberá ser resuelta por sus compañeros. En esta actividad los estudiantes toman el rol de facilitadores durante dos horas de todo el curso. Nosotros esperamos que los estudiantes alcancen (considerando el dominio cognitivo) el nivel jerárquico de síntesis y evaluación, de acuerdo a la Taxonomía de Bloom, [2].

Proponemos las siguientes recomendaciones para llevar a cabo una actividad excelente:

- Diseñar una práctica para que sus compañeros la desarrollen en un tiempo máximo de 50 minutos. La práctica diseñada debe tener una relación muy fuerte con la exposición realizada.
- Desarrollar una lista de cotejo para el desarrollo de la práctica. Se puede establecer algunas reglas para la lista de cotejo, ejemplo 20 % de puntualidad, etc.
- Cotejar con el profesor el diseño de la práctica y la lista de cotejo previamente al día que le toca realizar su intervención ante el grupo para recibir retroalimentación por parte del profesor. La experiencia ha mostrado que los estudiantes que no pasan con el profesor muchas veces no realizan un buen desempeño.
- Aplicar la lista de cotejo diseñada (pulida) para calificar a sus compañeros de clase una vez que la hayan terminado.
- Enviar las calificaciones obtenidas por sus compañeros a más tardar una clase después de la fecha de realización de la práctica al profesor.

Se le permite al equipo tener cierta libertad.

Independientemente de la lista de cotejo que los estudiantes diseñan es importante que los estudiantes que van a tener el rol de facilitadores sepan cómo van a ser evaluados, para ello es importante dar a conocer una lista de cotejo donde se contemplen aspectos como:

- Si la descripción de la práctica fue realizada usando html.
- Si la descripción de la práctica está siendo vinculada desde el material de exposición.
- Si la descripción de la práctica es clara y precisa.
- Si la descripción de la práctica utiliza diagramas o figuras para entenderse mejor. Si apoya a sus compañeros en dudas para realizar su práctica.
- Si la lista de cotejo que los alumnos diseñaron evalúa los puntos especificados en la descripción de la práctica.

Proporcionamos una posible lista de cotejo, la cual puede servir de guía. Puede descargarse de: <https://practiceDesign.pdf> Tanto para la actividad de la presentación oral como del diseño de la práctica es importante tomar en cuenta una Auto y Coevaluación que debe ser llenada por los mismos estudiantes.

## 5.3. EXAMEN FINAL

Dentro de la evaluación basada en competencias, el Saber es una competencia que los alumnos deben adquirir en el transcurso de un curso. De acuerdo a los dos primeros niveles del dominio cognitivo de Bloom [1], el conocimiento y la comprensión son habilidades básicas y muy importantes que un estudiante debe poseer, ya que los conceptos aprendidos sirven para relacionar nuevos conocimientos.

Uno de los instrumentos utilizados para identificar, en los estudiantes, la competencia del conocimiento es realizar una prueba, la cual puede ser práctica y/o teórica. Usualmente las pruebas prácticas te dan evidencia que el estudiante ha adquirido habilidades del Saber Hacer. El problema de su aplicación es saber si el tiempo que se le da al alumno para resolver la prueba es el adecuado, ya que algunos pueden terminar más rápido que otros. Las pruebas teóricas normalmente miden conocimiento, y es una evidencia si el alumno aprendió los conceptos, pero en ningún momento sabes si adquirió las habilidades para resolver ciertos problemas.

Independientemente del tipo de prueba aplicado al estudiante, sirve también para identificar las posibles debilidades que contengan los estudiantes y después trabajar con ellos para reforzarlos. El examen final se

compone de 5 secciones. Las primeras 4 relacionadas a las unidades descritas en;

## 6. PRÁCTICAS

### 6.1. PRÁCTICAS DEL LADO DEL CLIENTE

Prácticas enfocadas a entender HTML Y CSS (respectivamente HTML5 y CSS3): En la primera práctica, los estudiantes deben aprender:

- La estructura general de una página html.
- Pueden usar las etiquetas básicas para dar formato a las fuentes.
- Esta actividad le permite a los estudiantes entender sobre una correcta estructura de una página web. En una segunda práctica, los estudiantes:
  - Pueden usar (entre otras cosas) el uso de divs, inputs, forms, enlaces. Se puede empezar con el uso de CSS. En una tercera práctica, los estudiantes deben:
- Entender las diferentes formas de incluir estilos a una página web.
- Conocer el uso de selectores de CSS, el uso de ids y class.
- Conocer como dar posiciones, dimensiones, colores, etc. a las etiquetas de bloque y texto.
- Entender cómo aplicar diferentes tipos de estilos, no es importante si no lo memorizan, pero si es importante que sepan cómo hacerlo.

En una cuarta y quinta práctica (prácticas diseñadas por dos equipos de Estudiantes): uno enfocados a HTML5 y el otro a CSS3 (ver subsección 6.4 para más detalles).

En cada una de las prácticas anteriores los estudiantes pueden incluir más etiquetas que las que se requieren, pero es importante enfatizar que deben entender todas las etiquetas que ellos incluyan a sus prácticas. Una buena sugerencia es realizar la revisión de las diferentes actividades en público para realizar retroalimentación y quizás aprender otros tags usados por otros estudiantes. Con esto, los estudiantes aprenden de sus compañeros.

Prácticas para entender JavaScript y HTML DOM:

- En la primera práctica, los estudiantes deben aprender:
- Como ejecutar un simple código de JavaScript;
- Como llamar funciones desde eventos en las etiquetas (ej. onclick);
- Operadores lógicos y aritméticos.

En una segunda práctica, los estudiantes:

- Deben entender cómo acceder al DOM a través

de JavaScript;

- Pueden realizar diferentes operaciones entre JavaScript y el DOM de HTML.
- Para reforzar los conocimientos previos, también se incluyen dos prácticas más que deben ser diseñadas por equipos de estudiantes: un enfocado a JavaScript y el otro a la manipulación del DOM desde JavaScript (Ver subsección 6.4 para más detalles).

### 6.2. PRÁCTICAS DEL LADO DEL SERVIDOR

Estas prácticas van enfocadas para entender la programación del lado del servidor. Se sugiere tener como referencia a dos lenguajes de programación (en nuestro caso fueron JSP y PHP). Antes de que los estudiantes resuelvan las prácticas el profesor debe primeramente enseñar a los estudiantes los conceptos cliente-servidor.

- Con la primera práctica, el estudiante debe entender qué parte del script se ejecuta en el cliente y qué parte en el servidor. Así pues los estudiantes deben obtener los siguientes resultados:
- Conocer como instalar un servidor web y el path para incluir archivos que serán desplegados en un browser.
- Entender qué parte del script se ejecuta en el servidor y cuál en el cliente.
- Desarrollar una simple aplicación con ejecución en ambas partes: en el cliente y en el servidor.
- En una segunda práctica, los estudiantes:
- Pueden intensificar su conocimiento con respecto a diferentes instrucciones de programación del servidor, por ejemplo: instrucciones de ciclos.
- También pueden incrementar su nivel de aprendizaje en el lado del cliente (JavaScript) y también en el lado del servidor. Con esto entender las validaciones hechas para evitar latencia de datos.
- En una tercera práctica, los estudiantes:
- Deben entender el concepto de variables de sesión y cookies;
- Pueden incluir una práctica simple de autenticación (Ver trabajo de Francesco Maiorana [5] para profundizar en esta práctica);
- Entender el ciclo de vida de las variables de sesión y compararlas con las variables de funciones;
- En una cuarta práctica, los estudiantes:

- Deben entender cómo usar variables de sesión y acceso a Base de Datos;
- Pueden reforzar una práctica de autenticación con acceso a base de datos.
- Finalmente se propone dos prácticas diseñadas por equipos de estudiantes: uno enfocado a conocer la programación del lado del server y el otro enfocado a las variables de sesión y cookies. Para entender bien el paradigma, sería muy importante que los estudiantes trabajen con un lenguaje distinto al que proporciona los ejemplos el profesor, con esto los alumnos pueden ir comparando entre lenguajes de programación. 6.4).

### 6.3. PRÁCTICAS DE INTERACCIÓN CLIENTE-SERVIDOR

En estos días, es difícil pensar en una aplicación que no se desarrolle pensando en la facilidad de uso por parte del usuario final, siempre se busca proporcionar mejores formas de interacción, captura de datos eficiente y sin errores, el uso de tecnologías como AJAX permiten lograr una experiencia al usuario muy similar a las que antes solo se lograban con aplicaciones visuales de escritorio . Las prácticas sugeridas se enfocan en dar a entender el concepto de AJAX.

A veces los estudiantes piensan que el concepto de AJAX es dependiente del lenguaje de programación que se esté usando, es por eso que es bastante provechoso explicar el concepto de AJAX con prácticas en al menos dos lenguajes de programación (en nuestro caso JSP y PHP).

- En la primera práctica, se deja un desafío a los estudiantes (Aprendizaje Basado en Problemas): Cómo actualizar en tiempo real algunas partes de un sitio web, sin usar AJAX. En esta práctica el estudiante investiga por diferentes soluciones, es probable que algunos estudiantes aunque no conozcan el concepto de AJAX todavía, puedan dar soluciones usando AJAX, estos casos es importante identificarlo y aclararlo.
- Para una segunda y tercer práctica, el profesor debe haber ya explicado el concepto de AJAX, así pues los estudiantes simplemente tienen que resolver un simple problema donde el browser solicita al servidor alguna información que viene desde la base de datos, el browser debe presentar el resultado de forma asíncrona.

### 6.4. PRÁCTICAS DISEÑADAS POR LOS ESTUDIANTES

Las prácticas diseñadas por los estudiantes son una actividad complementaria descrita en 5.2. Esta actividad persigue que los estudiantes siempre tengan actividades diferentes con respecto a cursos previos. Algunas veces los estudiantes con tal de aprobar algunas actividades sólo buscan por la solución de cursos previos. Así pues, fomentando el diseño de prácticas por parte de los estudiantes, evitamos que los estudiantes sepan la solución anticipada de prácticas y con esto no sabrían cuál sería la práctica a resolver y se tendría mayor confianza en saber quiénes lo están desarrollando y quienes solo copiando.

## 7. PROYECTO

### 7.1. DISEÑO DE UN SIMULADOR DE SISTEMA OPERATIVO BASADO EN WEB

Se espera que con las especificaciones del proyecto inicial los estudiantes consagren su aprendizaje en los temas de HTML, CSS y JavaScript. Este proyecto contiene alto contenido en diseño, sobretudo en lograr la similitud en el diseño de las ventanas, ahí es donde los estudiantes confirman el uso de CSS. Algunos estudiantes podrán descargar imágenes de las ventanas que sean muy parecidas al Sistema Operativo que han seleccionado, y esto también es válido porque aprenden a manipular el uso de imágenes.

A continuación explicamos un ejemplo de las indicaciones que se le dan a los estudiantes y junto a ello los conocimientos que adquieren y los retos que presentan:

- El Sistema Operativo a replicar en diseño deberá mostrar el escritorio de inicio. Esto enfrenta varios retos de diseño considerando las diferentes propuestas que existen en Sistemas Operativos. Este apartado el alumno deberá aplicar conocimientos altos de HTML y CSS.
- Todos los Sistemas Operativos tienen un menú principal de opciones, acá se les solicita que coloquen un menú de opciones que contengan por lo menos las siguientes acciones:
  1. Identidad
  2. Calculadora personal; y
  3. Notas.
- Identidad: debe colocarlo en algún apartado

ingenioso del Sistema Operativo seleccionado y deberá mostrar los datos personales del desarrollador de la página, el logo de la universidad, materia, y nombre del profesor.

- Calculadora: cuando se dé clic en esta opción deberá aparecer una calculadora similar en diseño al Sistema Operativo elegido, y deberá calcular operaciones aritméticas como suma, resta, multiplicación, división y residuo (mod). Esto involucra consolidar los conocimientos en JavaScript, html y CSS.
- Notas: cuando se da clic a esta opción, deberá aparecer un sistema de notas (crear y borrar notas). Como aún no se ha visto los temas que implican al servidor no se permite persistencia de datos. Este apartado permite al alumno consolidar sus conocimientos entre JavaScript y su acceso al HTML DOM.
- Algo importante que se le pide al alumno es que si el sistema operativo que ha seleccionado no cubre los aspectos antes mencionados, deberá realizar algo adicional para cumplir con los puntos especificados.

## 7.2. PROYECTO FINAL: SIMULADOR DE SISTEMA OPERATIVO

El proyecto final consistirá en varios aspectos descritos a continuación:

Continuar el proyecto inicial que se inició en la parte del diseño 7.1;

- Desarrollar una aplicación web usando cualquiera de los lenguajes de programación de lado del servidor vistos en clase (en nuestro caso fue JSP o PHP), pero es importante que el estudiante tome una decisión con anticipación. Con esto logramos que cada estudiante tenga diferente sistema operativo que diseñar e inclusive con algún lenguaje de su propia preferencia y no necesariamente un lenguaje visto en clase.
- El sistema operativo debe tener dos tipos de usuarios: a) Administrador y b) usuario general;
- El administrador puede agregar otros usuarios;
- El usuario general puede navegar entre diferentes opciones (como son especificadas en la Sección 7.1 and a continuación).
- La única opción para acceder al simulador del Sistema Operativo debe ser a través de un control de acceso (validado como fue especificado en el item anterior).

- Implementar el uso de catálogos con acceso a base de datos, extendiendo el uso de un sistema de notas, donde el usuario podría agregar, borrar, actualizar y listar notas, pero esta vez implementando persistencia de datos.
- Otro aspecto importante es el uso de archivos, en este caso se ha dejado el desarrollo de un editor de texto con algunas capacidades (al menos, cambiar fuente, cambiar tamaño de fuente, e implementar las opciones "hacer" "deshacer").

## 8. RESULTADOS

### 8.1. ÍNDICE DE APROBACIÓN Y REPROBACIÓN

El método de evaluación descrito en este artículo lo hemos aplicado desde hace 7 años con un aproximado de 9 grupos distintos, 6 en la Universidad Politécnica de Chiapas (UPChiapas) y 3 en el Instituto Tecnológico de Tuxtla Gutiérrez (ITTG), ver la Tabla 2 para un resumen estadístico.

Tabla 2 Estadística de alumnos que han llevado la metodología de trabajo

Descripción	Núm. Alumnos	Núm. Aprobados	No Aprobados	Escuela	% Aprob.
1a Generación	23	14	9	UPChiapas	61 %
3a Generación	34	23	11	UPChiapas	68 %
4a Generación	24	14	10	UPChiapas	58 %
5a Generación	36	22	14	UPChiapas	61 %
7a Generación	40	28	12	UPChiapas	70 %
8a Generación	30	21	9	UPChiapas	70 %
Grupo A2017	18	14	4	ITTG	78 %
Grupo B2017	27	26	1	ITTG	96 %
Grupo C2017	30	22	8	ITTG	73 %

Es importante aclarar que en cada grupo se van haciendo cambios graduales y de ahí posiblemente la mejora en cuestión de porcentajes de aprobación. Por ejemplo los últimos grupos han sido aplicados en el ITTG, donde podemos observar mejor índice de aprobación. Aunque debemos aclarar que los cursos impartidos en

el ITTG tienen mayor duración que en la UPChiapas y en el semestre que se imparte esta materia en el ITTG ya han llevado tópicos de Base de Datos, en cambio en la UPChiapas aún no y se tiene que impartir algunas nociones básicas de SQL dentro de la materia. De ahí que la reflexión a la cual hemos llegado es que, para el caso de Universidades Politécnicas, es importante dividir el curso en dos como lo sugiere Ye Diana Wang et.al. en [9] y Xusheng Wang[8]. Para el caso del ITTG que ya tienen dos cursos enfocados al desarrollo de aplicaciones web se recomienda pasar el curso de Programación Web a un semestre previo y el curso de Tópicos Avanzados en Tecnologías Web en un segundo curso pero en un siguiente semestre (es decir el último en este caso). Realizándose estos cambios entonces podría ser posible incluir metodologías como lo sugiere Harrigier y Woods [3] donde propone un método de enseñanza basado en Desarrollo de Sitios Web para negocios locales, incluyendo un metodología de desarrollo de software basado en web. Otro trabajo que sugiere algo similar es el de Margaret et.al. [6] donde propone el desarrollo de un portal con equipos de dos o tres estudiantes.

## 8.2. APRECIACIÓN DE LOS ESTUDIANTES DE LA METODOLOGÍA

Al aplicar esta metodología de trabajo, se han aplicado encuestas, auto y coevaluaciones en cada una de las actividades, acá mostramos algunas relevantes y sobre todo aquellas mencionadas en secciones 5.1 y 5.2, que representan una parte fundamental de la generación del conocimiento del curso. Además que ha servido como experiencia para mejorar la propuesta metodológica, ver Figura 2.

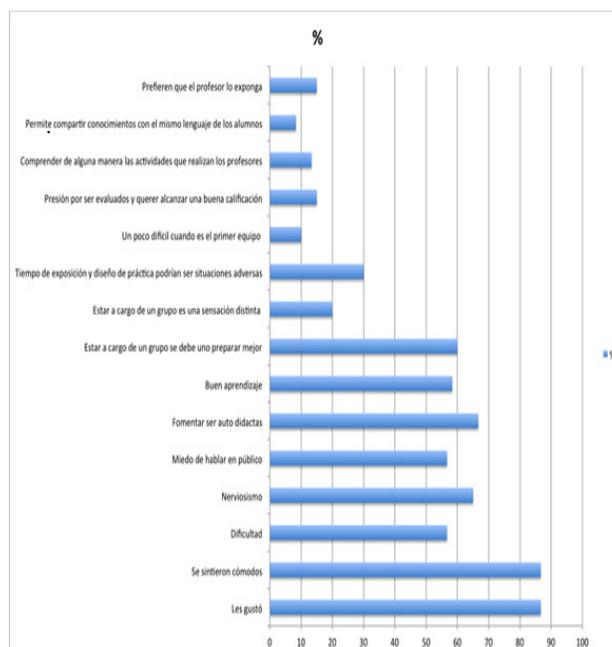


Figura 2 Gráfico sobre la aplicación de la metodología en los estudiantes

## 9. CONCLUSIONES

En este artículo hemos presentado un método de enseñanza para cursos presenciales en la enseñanza de Programación Web basado en Evidencias de Aprendizaje. A lo largo del escrito, hemos compartido nuestra experiencia y conocimiento en la aplicación de la metodología y sus actividades.

El método de evaluación descrito en este artículo lo hemos aplicado desde hace 7 años con un aproximado de 9 grupos distintos, 6 en la Universidad Politécnica de Chiapas (UPChiapas) y 3 en el Instituto Tecnológico de Tuxtla Gutiérrez (ITTG), ver la Tabla 2 para un resumen estadístico. En cada año hemos realizado cambios graduales y también hemos identificado mejores niveles de aprobación. En el apartado de resultados hemos dado algunas recomendaciones para los dos subsistemas ITTG y UPChiapas que consideramos muy apropiados para mejorar el plan curricular de estas carreras universitarias.

Este artículo consideramos puede ser usado por profesores quienes están buscando una estrategia de enseñanza-aprendizaje en cursos avanzados de Tecnologías Web. Esta estrategia es adecuada siempre y cuando el profesor tenga enfrente a estudiantes de grados avanzados. Esta estrategia inclusive puede aplicarse en otras materias, pero es recomendable que cuando existan temas (o paradigmas) nuevos, el profesor

introduzca primero la explicación del tema, ya que la experiencia ha demostrado que siempre un paradigma nuevo es mejor explicado por el titular (que ya tiene experiencia).

Adicionalmente consideramos que aunque el fin del artículo no es comprobar la hipótesis planteada en la Introducción, de alguna manera la hemos validado ya que generación tras generación hemos construido materiales dinámicos que permiten a estudiantes resolver prácticas distintas cada año. Además, esto ha demandado que cada año se introduzcan temas nuevos tanto por parte de los facilitadores como de los estudiantes.

## REFERENCIAS

- [1] Lorin W. A. and David R. Krathwohl, editors. A Taxonomy for Learning, Teaching, and Assessing. A Revision of Bloom's Taxonomy of Educational Objectives. Allyn & Bacon, New York, 2 edition, December 2001.
- [2] B. S. Bloom, M. B. Engelhart, E. J. Furst, W. H. Hill, and D. R. Krathwohl. Taxonomy of educational objectives. The classification of educational goals. Handbook 1: Cognitive domain. Longmans Green, New York, 1956.
- [3] Alka R. H. and Denise M. Woods. A Structured Approach To Teaching Web Development. Distributed by ERIC Clearinghouse [Washington, D.C.], 2001.
- [4] Li Y. and Phelps C. G. and professional tools used when teaching web programming. J. Comput. Sci. Coll., 26(5):116{121, May 2011.
- [5] Maiorana F. Teaching web programming - an approach rooted in database principles. In CSEDU: 6th International Conference on Computer Supported Education, pages 508 { 5013, 2014.
- [6] R. E. Margaret, A. M. Vinod, and Tejonidhi M. R. A practical approach to teach web programming course. Journal of Engineering Education Transformations, 0(0), 2016.
- [7] Theodore. H. Kaskalis Stelios Xinogalos. The challenges of teaching web programming - literature review and proposed guidelines. In Krempels KarlHeinz Cordeiro, José, editor, WEBIST 2012: - 8th International Conference on Web Information Systems and Technologies, volume 1740 of Lecture Notes in Business Information Processing, pages 206{212, Porto, Portugal, 2012. Springer-Verlag.
- [8] Wang X. Design, develop and teach the second web programming course in computer science curriculum. J. Comput. Sci. Coll., 29(4):52{59, April 2014.
- [9] Ye D. W. and Zahadat N. Teaching web development in the web 2.0 era. In Proceedings of the 10th ACM Conference on SIG-information Technology Education, SIGITE '09, pages 80-86, New York, NY, USA, 2009. ACM.

SEMBLANZA



López Pimentel Juan Carlos  
 Doctorado en Ciencias Computacionales y Maestría en Ciencias de la Computación, ambos por parte del Tecnológico de Monterrey, CEM. Se graduó en el 2001 con mención honorífica de la carrera de Ingeniería en Sistemas Computacionales del Tecnológico de Tuxtla Gutiérrez. Ha participado en varios cursos y visitas científicas en el extranjero. Vasta experiencia en Docencia, Investigación y coordinación de proyectos de investigación. Ha impartido cursos en varias Universidades en niveles de licenciatura, maestría y doctorado. Alrededor de 20 artículos publicados. Ha participado en 10 proyectos de Investigación y Desarrollo de Software de los cuáles ha generado considerable producción académica. Ha sido evaluador in situ del CONACyT. Acreedor de varios reconocimientos académicos y económicos los que le distinguen son el nombramiento de formar parte del Sistema Nacional de Investigadores, Nivel C y también parte del Sistema Estatal de Investigadores Nivel III Honorífico (Estado de Chiapas); reconocimiento a Perfil Deseable. Actualmente es Profesor del Tecnológico Nacional de México, Campus Tuxtla Gutiérrez.



María Candelaria Gutiérrez Gómez  
 Ingeniera en Sistemas Computacionales, Maestra en Administración, Maestra en Administración de Tecnologías de Información y Doctora en Administración. Actualmente es profesora en el Programa de Pregrado de la Licenciatura en ingeniería en Sistemas Computacionales del Tecnológico Nacional de México, Campus Tuxtla Gutiérrez y en el Programa de Posgrado de la Maestría en Administración de la Facultad de Contaduría y Administración Campus I de la Universidad Autónoma de Chiapas. Está certificada por el Consejo Nacional de Normalización y Certificación de Competencias Laborales (CONOCER) (2015-2017) Norma EC 0217 Impartición de cursos de formación del capital humano de manera presencial grupal. Ha participado en la elaboración de reactivos para el examen EGEL-Informática. Recibió el Reconocimiento de Excelencia otorgado por el ITESM al Mejor Promedio obtenido durante los estudios de Maestría en Tecnologías de Información en el 2014 y es miembro fundador del Colegio de Educación a Distancia del Sistema Nacional de Educación Superior Tecnológica.



Imelda Valles López  
 Desde 1991, profesora del área de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Tuxtla Gutiérrez. Maestro en Administración por el I.T.T.G., en 2001 e Ingeniero en Sistemas Electrónicos por el ITESM en 1990. Docente de las materias de Redes, Teoría Matemática de la Computación y Compiladores. Logros importantes:

- Fundador de la línea de trabajo "Desarrollo de software para hablantes en lenguas nativas del estado de Chiapas" (2010).
- Líder de la línea de investigación "Cómputo Educativo";

desde 2012.

- Integrante del cuerpo académico "Tecnología computacional para el desarrollo regional", ITTUXG-CA-4. Desde 2011.
- Profesor de tiempo completo con perfil deseable (2013-2019).
- Miembro del Sistema Estatal de Investigadores. Investigador Tecnólogo Nivel I desde 2012.



Francisco de Jesús Suárez Ruiz  
 Profesor en el área de Ingeniería en Sistemas Computacionales, es Maestro en Ciencias Computacionales egresado de la Universidad Pablo Guardado Chavez. Su experiencia en docencia es en el área de programación, matemáticas discretas, redes de computadoras y teoría de compiladores. También es profesor en el Colegio de Educación Profesional Técnica en las área de Redes de Computadoras, Sistemas Operativos y Matemáticas Discretas.



Alicia González Laguna  
 Tiene una maestría en Comunicación y Tecnologías educativas por el Instituto Latinoamericano de la Comunicación educativa (ILCE) y es Licenciada en Computación por la Universidad Autónoma de Tabasco (UJAT). Desde 1996, es docente en la carrera de Ingeniería en Sistemas Computacionales de Instituto Tecnológico de Tuxtla Gutiérrez, impartiendo actualmente las materias de Fundamentos de Programación y Programación Orientada a Objetos, también ha dirigido y participado en diversos proyectos de su carrera. Sus áreas de interés son: Programación y computación educativa. La Mra. González ha sido revisora de proyectos tanto de tesis como de residencias. Ha publicado artículos en diversos congresos y revistas de su área.



Octavio A. Ríos Tercero  
 Maestro en Ciencias en Ciencias Computacionales egresado del Centro Nacional de Investigación y Desarrollo Tecnológico CENIDET. Su experiencia en docencia es en el área de ingeniería de software. Profesor de tiempo completo y Actual Jefe de Proyectos de Investigación en el Depto. de Sistemas y Computación del Tecnológico Nacional de México campus Tuxtla Gutiérrez.



Profesor de tiempo completo del Tecnológico Nacional de México, Campus Tuxtla. Licenciatura en Sistemas Computacionales y Maestría en Comercio Electrónico por el Tecnológico de Monterrey. Doctorado en Sistemas Computacionales por la Universidad del Sur. Su área de especialidad son las redes de computadoras y bases de datos.