

Modelado y Programación 2017-1:

Práctica 13, Tecnologías Web para Back-End y Web 2.0.

Cinthia Rodriguez Maya (cinthia.rodriguez@ciencias.unam.mx)
María del Carmen Sánchez Almanza (carmensanchez@ciencias.unam.mx)
Ricchy Alaín Pérez Chevanier (alain.chevanier@ciencias.unam.mx)

Fecha de Publicación: 23 de Noviembre de 2016
Fecha de Entrega: 30 de Noviembre de 2016

1 Introducción

1.1 Desarrollo Back-End

¿Qué hace posible el funcionamiento completo de un sitio web? ¿Dónde se encuentra toda la información almacenada? En la respuesta a estas preguntas es donde surge el Back-End. El Back-End de un sitio o aplicación web consiste de un servidor, una aplicación y un almacén de información. Un desarrollador de Back-End construye y mantiene la tecnología que soporta esos componentes, que de manera conjunta, habilitan la consulta y persistencia de la información que alimentan a un sitio o aplicación web.

1.2 Habilidades y Herramientas

Con la finalidad de que un servidor, una aplicación y un almacén de datos se comuniquen entre sí, el desarrollador tiene que utilizar lenguajes que se ejecuten del lado del servidor, como son PHP, Ruby, Python, Java, C# e incluso Javascript, etc. También almacenes de datos como son MySQL, Oracle, MariaDB, SQLServer, MongoDB, CouchBase, etc que sirven para buscar, persistir y alterar la información con la que interactúa el usuario final por medio del Front-End. Así como también servidores http de aplicaciones como son Apache, Nginx, Tomcat, glassfish, etc.

Uno de los objetivos principales de la comunidad de desarrolladores de software en el mundo ha sido facilitar y agilizar su labor a través del tiempo, por lo cual han sido creados diferentes frameworks de desarrollo en una gran variedad de lenguajes de programación, por ejemplo para PHP existen Zend, Laravel, CodeIgniter, etc; para Ruby existe Rails; para Java existen Spring MVC y Struts; para Groovy existe Grails; etc.

Otra tipo de herramientas muy importantes que han surgido para facilitar el trabajo de los desarrolladores son los sistemas de control de versiones como Subversion, Mercurial y Git, cuyo objetivo principal es ayudarnos a identificar el historial de cambios de los proyectos en los que los que trabajamos.

Finalmente parte importante del desarrollo de Back-End incluye el despliegue de aplicaciones a un servidor de aplicaciones. Para ellos han surgido varias herramientas como Vagrant y Docker.

1.3 AJAX

Ajax (Asynchronous Javascript and XML por sus siglas en inglés) es un conjunto de técnicas

de desarrollo web que utilizan diversas tecnologías web client-side para crear llamadas asíncronas en nuestra aplicación web. Con Ajax, las aplicaciones web mandan y reciben información hacia y desde el servidor asíncronamente sin interferir con el despliegue y comportamiento de la vista en cuestión mediante el desacoplamiento de la capa de intercambio de información y la capa de presentación de la misma. En general Ajax permite transmitir contenido dinámicamente a una aplicación web sin necesidad de recargarla por completo. En la práctica, las implementaciones modernas sustituyen el uso de XML por JSON dada la ventaja de ser nativo en javascript. El DOM es accedido y modificado para desplegar dinámicamente la información presentada. Javascript y el objeto XMLHttpRequest proveen un método para intercambiar información asíncronamente entre el navegador y el servidor para evitar la recarga completa del sitio.

2 Actividades

2.1 Descripción

Tomando con referencia la agenda telefónica que construiste en la práctica anterior, agrega la siguiente funcionalidad:

1. Filtrado de contactos: Mediante una caja de texto filtra los contactos cuyos nombres o apellidos coincidan parcial o enteramente con lo escrito en esta (el filtrado se realiza de manera automática conforme el usuario escribe en la caja de texto).
2. Persistir los contactos creados por medio de la interfaz en una base de datos.
3. Eliminar de la base de datos los contactos eliminados por medio de la interfaz.

Si utilizaste AngularJS para realizar la práctica anterior, te recomendamos añadir las bibliotecas de javascript Sugar.js o Underscore.js para manipular fácilmente el arreglo que representa la lista de contactos.

Para construir el código que atenderá las peticiones que realizarás por medio del cliente web, puedes utilizar PHP para manejar la conexión a la base de datos, y para realizar operaciones sobre ella (agregar elementos y eliminarlos). Posiblemente sea un poco más simple utilizar una base de datos no relacional para este ejercicio (Si utilizas MongoDB te podemos orientar. También puedes utilizar una base de datos serverless como SQLite).

2.2 Puntos extra

Si utilizas GIT para realizar esta práctica y tu repositorio tiene al menos 3 commits, tendrás un punto extra.

3 Entrega

No olvides incluir en tu entrega todas las dependencias de tu implementación como pueden ser angular.min.js, bootstrap.min.js, etc.

Incluye la información necesaria para crear el esquema de base de datos que utilizaste para tu solución (aplica para relacionales y no relacionales).