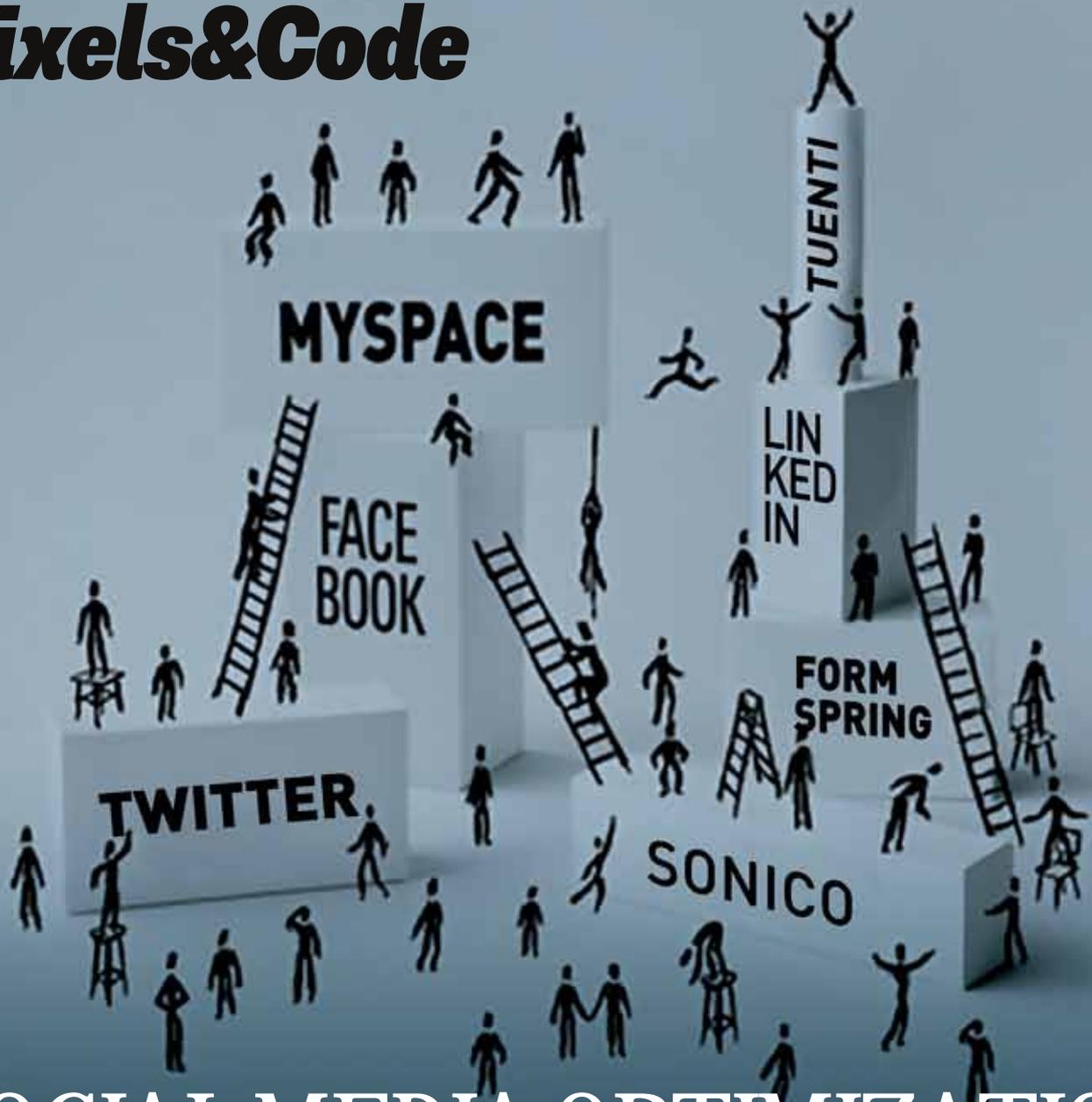


MAYO 2011 > N° 02

GEOLOCALIZACIÓN
EN APPS PARA IOS

SITIOS OPTIMIZADOS
PARA NAVEGACIÓN MÓVIL

Pixels&Code



SOCIAL MEDIA OPTIMIZATION

CÓMO PUBLICITAR TU SITIO EN REDES SOCIALES Y COMUNIDADES ONLINE

Plataforma para Resellers

¡El negocio lo hacés vos!

Tu propio negocio de hosting con todo lo que necesitás.
Con más de 8 años de experiencia, entendemos tus necesidades. Por eso creamos una plataforma con productos y herramientas para que tu negocio crezca con la mayor rentabilidad.

www.dattatec.com/resellers





Connecting to server...

Por Débora Orué

Muy complacidos con el recibimiento del primer número de Pixels&Code, aquí estamos retomando nuestro contacto mensual con todos ustedes.

Escuchamos atentamente todas las sugerencias que nos fueron acercando referentes a las temáticas que quieren ver desarrolladas en esta publicación. En la tapa de este número verán la respuesta al tema

más solicitado: social media. Y unas páginas más adelante otra temática indispensable, tan nueva como apasionante: el desarrollo de aplicaciones para móviles. En esta edición vamos a jugar un poco con la geolocalización en dispositivos de Apple.

También notamos mucho interés por HTML5/CSS 3, Javascript y Android, así que cuenten con esos temas para futuras ediciones.

Claro está que esto no termina aquí. Recién estamos empezando. Recién nos estamos conociendo. Así que la propuesta sigue abierta para que nos escriban y nos cuenten qué quieren leer en las páginas de Pixels&Code. Nuestro mail de contacto es lectores@pixelscode.com y queda a su disposición para que nos hagan llegar nuevas críticas y sugerencias. Los dejo con el segundo número de P&C. ¡Que lo disfruten! **P**



“
dattatec.com
Tu Hosting hecho Simple!

JEFA DE REDACCIÓN



Débora Orué

COLUMNISTAS



*Adrián Ortiz
Cristian Hernán Gaitano Ornia
Gabriel Walter Gaitano Ornia
Juan Gutmann
Roxana Miguel
Natalia Cantero*



DISEÑO EDITORIAL Y CREATIVIDAD

www.ampersandgroup.com.ar

REDACCIÓN

lectores@pixelscode.com

COMERCIALIZACIÓN

*Dattatec.com SRL - Córdoba 3753, Rosario, Santa Fe
www.dattatec.com*

DATOS DE CONTACTO

*Dirección Comercial:
publicidad@pixelscode.com*

Las opiniones expresadas en los artículos son exclusiva responsabilidad de sus autores y no coinciden necesariamente con la opinión de Dattatec.com SRL.

sumario

NOTA CENTRAL 32 // PROMOCIÓN WEB

Aprovechando
las redes sociales
como herramientas
de difusión



6 // HERRAMIENTAS PARA EL DISEÑADOR

El significado de los colores en la web

10 // BASES DE DATOS

Introducción a las bases de datos relacionales (II)

16 // APLICACIONES MÓVILES

Geolocalización en aplicaciones móviles

22 // HERRAMIENTAS PARA EL PROGRAMADOR

*Frameworks en PHP:
Codelnigter, una gran opción*

36 // MULTIMEDIA

Extensiones para Firefox indispensables para desarrolladores web

40 // ALGORITMOS

Algoritmos Insertion sort

42 // HERRAMIENTAS PARA EL DISEÑADOR

*Pensar en grande
para dispositivos chicos*





APUNTES DE DISEÑO: COLOR

El significado de los colores en la Web

La pantalla es un lienzo para los diseñadores web que a la hora de enfrentar una paleta de colores deben decidir lo correcto para que el resultado genere las sensaciones buscadas. Aquí sabremos el por qué de cada color para aprender a usarlos con más inteligencia.

Roxana Miguel

Roxana.miguel@pixelscode.com

@roxanamiguel



Hay un ejercicio sencillo para empezar este camino hacia el descubrimiento de los colores. Busca tres páginas en Internet al azar y escribe en pocas palabras la primera impresión que ésta te genera al verla. No te detengas a leer su contenido, sólo decide cuál te agradó más, cuál te genera dudas y cuál te desagradó al respecto de las otras dos. De un universo de tres ejemplos, verás que los colores y la organización de los objetos se

impondrá en tu vista y logrará que tengas distintas sensaciones: la mejor, la que genera dudas y la peor.

Está comprobado que son los primeros segundos los que atrapan a un usuario frente a un website al que llegan azarosamente. Aún cuando recurren a éste motivados por un interés, el primer impacto define y atrapa. Para generar una buena impresión el uso apropiado de la paleta de colores es fundamental ya

que evoca sensaciones automáticas e inevitables.

En el mundo de la comercialización se entiende que la armonía de una Web puede atraer o repeler, por lo que bien manejada esa energía psicológica de los colores puede lograr resultados muy positivos transportando la sensación correcta con una combinación de colores apropiada. El desafío de los diseñadores web será trabajar con una paleta de unos

cuantos colores que sirvan para expresar un mensaje claro.

ENERGÍA VIBRATORIA

El origen del color proviene de la intervención de la luz. En la oscuridad todo tiene una tonalidad que aparenta ser uniforme teñida por las sombras. Un simple hilo de luz nos abre un camino sobre los destellos que diferencian una cosa de otra. Es por esto que el color, como espectro lumínico, es al mismo tiempo energía vibratoria.

Toda energía afecta al ser humano de distintas formas, es por ello que quienes trabajan con el color deben aprender a conseguir las asociaciones correctas para el producto que ofrecen. Comunicar por medio de un site implica saber de palabras, imágenes y colores. Estos son los lenguajes para lograr una buena comunicación en una superficie que en lugar de papel está compuesta por bytes.

La cromoterapia, utilizada como parte de la psicología de los colores, versa sobre la utilización de la energía de la luz en diferentes frecuencias y hasta con fines curativos. Esta disciplina no está comprobada como una ciencia en sí misma, sin embargo, nos aporta datos importantes en cuanto a qué recibe el ojo humano del universo de luz.

El color es captado como una señal nerviosa por el cerebro e interpretado según las diferentes longitudes de onda de un espectro electromagnético. El ser humano es capaz de

captar una pequeña porción de un espectro visible que comprende longitudes de onda desde los 380 nm hasta los 780 nm (1nm = 1 nanómetro = 0,000001 mm). Cada una de estas medidas es percibida como un color diferente, por lo que en la retina entra una sucesión de longitudes de onda correspondiente al total de lo que normalmente observamos.

Como podemos apreciar hasta un millón de colores, nos valdremos de las células llamadas bastones y conos para procesar la entrada de luz. La información de luz y color se codificará mediante impulsos eléctricos y se transmitirá por medio del nervio óptico, provocando que el cerebro genere reacciones y en consecuencia libere hormonas que modifique el metabolismo o la conducta.

Por todo esto es necesario que a la hora de diseñar un web site se preste gran atención a la paleta de colores que se utilizará, porque por medio de la energía vibratoria de cada color lograremos transmitir a la perfección una sensación y un mensaje.

EL SIGNIFICADO DE LOS COLORES

Existe mucha bibliografía acerca de lo que simboliza cada color, sin embargo, en el universo de Internet siguen apareciendo sitios que parecen desconocer algunas reglas básicas. Aquí apuntaremos características de la paleta que te ayudarán para elegir una buena combinación según lo que quieras transmitir:

Blanco: Está asociado con la luz y con características como la bondad, inocencia y virginidad. Se lo considera un color perfecto y puro. Significa seguridad y limpieza. Tiene una connotación positiva. Puede representar un inicio afortunado. En heráldica, el blanco representa fe y pureza. En la promoción de productos de alta tecnología, puede utilizarse para comunicar simplicidad. También se lo asocia con hospitales y distintos ámbitos de la sanidad, con productos bajos en calorías y lácteos.

Amarillo: Simboliza la luz del sol, alegría, felicidad y energía. Sugiere el efecto de entrar en calor, estimula la actividad mental y genera energía muscular. Con frecuencia se le asocia a la comida. El amarillo puro y brillante es un reclamo de atención. En heráldica el amarillo representa honor y lealtad. Es recomendable utilizarlo para provocar sensaciones agradables, para promocionar productos infantiles, de ocio y para destacar los aspectos más importantes de una página web.

Naranja: Combina la energía del rojo con la felicidad del amarillo. Está asociado con las sensaciones de alegría, con el sol brillante y el trópico. También representa el entusiasmo, la felicidad, la atracción, la creatividad, la determinación, el éxito, el ánimo y el estímulo. Produce la sensación de mayor aporte de oxígeno al cerebro, vigorizando y estimulando la actividad mental. Es muy útil para sitios de temáticas jóvenes, alimentación sana y al estímulo del apetito. En heráldica



representa la fortaleza y la resistencia.

Rojo: Es el color del fuego y el de la sangre, por lo que se le asocia al peligro, la guerra, la energía, la fortaleza, la determinación, la pasión, el deseo y el amor. Mejora el metabolismo humano, aumenta el ritmo respiratorio y eleva la presión sanguínea. Es muy recomendable para inducir a las personas a tomar decisiones rápidas durante su estancia en un sitio web, también para anuncios de autos, motos, bebidas energéticas, juegos, deportes y actividades de riesgo. En heráldica el rojo simboliza valor y coraje. Es un color muy utilizado en las banderas de muchos países.

Azul: Es el color del cielo y del mar, por lo que se suele asociar con la estabilidad y la profundidad. Representa la lealtad, la confianza, la sabiduría, la inteligencia, la fe, la verdad y el cielo eterno. Se le considera un color beneficioso tanto para el cuerpo como para la mente. Retarda el metabolismo y produce un efecto relajante. Es un color fuertemente ligado a la tranquilidad y a la calma. En heráldica el azul simboliza la

sinceridad y la piedad. Es muy adecuado para presentar productos relacionados con la limpieza (personal, hogar o industrial), líneas aéreas, acondicionadores de aire, paracaidismo, cruceros, vacaciones, agua mineral, parques acuáticos y balnearios.

Verde: Es el color de la naturaleza por excelencia. Representa armonía, crecimiento, exuberancia, fertilidad y frescura. Tiene una fuerte relación a nivel emocional con la seguridad, también una correspondencia social con el dinero. Es el color más relajante para el ojo humano y puede ayudar a mejorar la vista. En heráldica el verde representa el crecimiento y la esperanza. Es recomendable utilizar el verde asociado a productos médicos, jardinería, turismo rural, actividades al aire libre o productos ecológicos.

Negro: Representa el poder, la elegancia, la formalidad, la muerte y el misterio. Es el color más enigmático y se asocia al miedo y a lo desconocido, también a la autoridad, fortaleza, prestigio y seriedad. En heráldica el negro representa el dolor y la pena. En una página web puede dar imagen de

elegancia, y aumenta la sensación de profundidad y perspectiva. Es típico su uso en museos, galerías o colecciones de fotos on-line, debido a que resalta los demás colores. Contrasta muy bien con los brillantes.

COLORES NEGATIVOS

Así como los mencionados anteriormente tienen más particularidades positivas que negativas, existen aquellos que en exceso producen las peores sensaciones. Si no quieres cometer errores presta atención:

Lavanda: Está asociado al equilibrio, sin embargo en abundancia produce cansancio y desorientación.

Amarillo: Prueba mirar durante unos segundos un sitio donde prima un amarillo intenso. Terminará por agotarte y esto es porque genera demasiada actividad mental. Por otro lado, aunque el derivado color oro es elegante también es demasiado fuerte para algunas personas.

Naranja: Representa la energía, tanto así que aumenta la ansiedad. Con el rojo intenso sucede lo mismo, logrando agitación y tensión.

Púrpura: Aunque compensado genera serenidad, en exceso es la receta para los pensamientos negativos.

Azul: Demasiado azul provoca depresión, aflicción y pesadumbre. Sería apropiado evitarlo para productos alimenticios y relacionados con la cocina en general, porque es un supresor del apetito.

Verde: Al igual que el púrpura, desmedidamente crea energías negativas.

Negro: Aunque no parezca necesario aclararlo, en exceso es distante e intimidatorio y dificulta la lectura cuando hay texto.

(Fuente: <http://www.webusable.com/coloursMean.htm>)

MALAS DECISIONES

No todas las combinaciones generan un aspecto positivo. Aquí veremos una lista de errores comunes que terminan por espantar a los visitantes:

- **Fondo oscuro y textos claros:** La elección de combinaciones con-

trastantes normalmente dificulta la llegada de los navegantes.

- **Fondo oscuro con texto oscuro:**

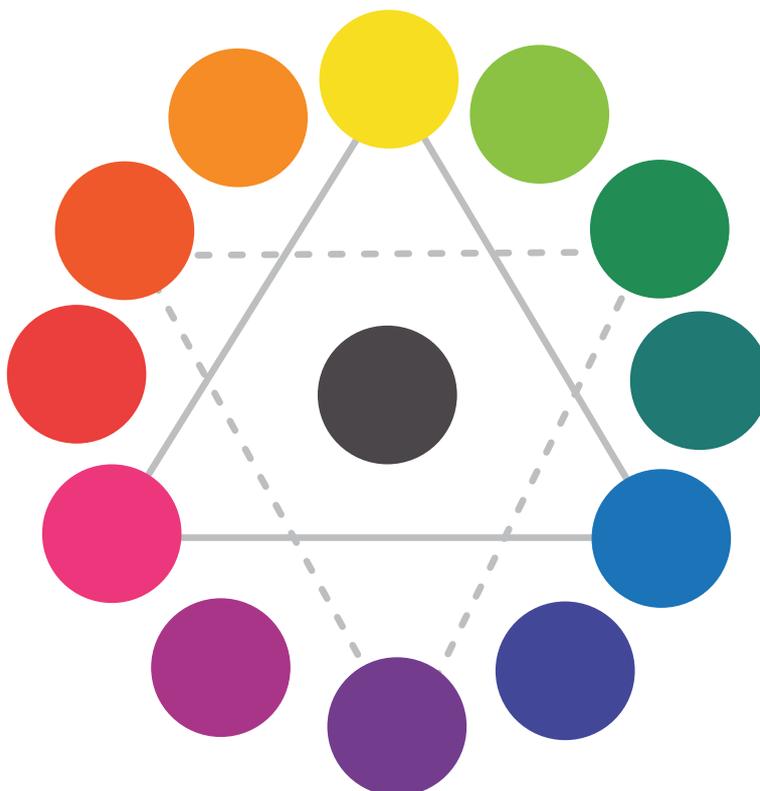
Aunque no sean de la misma gama de colores, el visitante deberá resaltar los textos que quiera leer para acceder a ellos con más facilidad y esto invita al usuario a buscar otro sitio más legible.

- **Imagen de fondo:** Es una decisión arriesgada ya que la imagen puede definir perfectamente a la compañía pero al mismo tiempo entorpecer el resto de la información del sitio, además de demorar la carga.

La forma más sencilla de combinar los colores es tomar el círculo cromático y aplicar los opuestos en un diseño dependiendo del significado

de cada uno de ellos y la propuesta de la compañía. Seguramente el resultado será exitoso. ¡Pruébalo!

La percepción del color termina por ser un indicador fundamental para un diseñador web que puese a crear un site que impresione a sus clientes quizás sólo deba bastarse de una simple paleta de tonalidades que reafirme un mensaje directo y que genere una sensación positiva. El mejor consejo en estos casos es lograr una armonía en la impresión del ojo humano, independientemente del cliente. Seguramente éste verá con agrado que el resultado es, a simple vista, la mejor luz puesta sobre su empresa. **P**



Está comprobado que son los primeros segundos los que atrapan a un usuario frente a un website al que llegan azarosamente.”



Introducción a las bases de datos relacionales (II)

Juan Gutmann
juan.gutmann@pixelscode.com

En la primera entrega de esta serie de notas, en las que nos proponemos difundir los “conceptos base” que permitirán al desarrollador de software hacer un buen uso de los motores de bases de datos relacionales, narramos la creación del “Modelo Relacional” por parte de Edgar F. Codd, considerado el padre de este vasto terreno de la ingeniería informática. Junto con los “doce mandamientos” que proponen una serie de condiciones que un motor de base de datos debe cumplir para ser considerado relacional, Codd concibió la “Normalización” de los datos almacenados por el motor. Con el término normalización llamamos en nuestros días a un conjunto de reglas que deben aplicarse al modelar un conjunto de datos que desean administrarse mediante un modelo relacional. Dichas reglas tienen múltiples objetivos, siendo los principales mantener la integridad de la información y de las relaciones que existan entre las distintas entidades,

y evitar la redundancia de datos, para optimizar los recursos utilizados por el motor, principalmente memoria y espacio en las unidades de almacenamiento. En esta ocasión, vamos a adentrarnos en las características de las reglas creadas por Codd para mantener “normalizada” una base de datos.

FORMAS NORMALES

Codd concibió originalmente tres formas normales, llamadas primera forma normal (1NF), segunda forma normal (2NF) y tercera forma normal (3NF), respectivamente. Dichas formas normales no se aplican a la base de datos como un todo, sino a las entidades y relaciones individualmente, que en el modelo relacional se conforman como “tablas”, según veremos más adelante. Si afirmamos que una base de datos cumple en su totalidad con la primera forma normal, es porque absolutamente todas las tablas contenidas en ella cumplen con dicha forma. Cuanto

más alto sea el número de forma normal que cumple una tabla, menor será la posibilidad de que se generen inconsistencias en la información almacenada en ella, o en los métodos que se hayan diseñado o se generen a futuro para modificar dicha información. Cada forma normal debe cumplir con las reglas que la definen, pero también con las que definen a las formas normales de “menor rango”. Es decir, para que se considere que una tabla está en 2NF, también debe cumplir con las reglas enunciadas en la 1NF.

PRIMERA FORMA NORMAL

La definición formal de la 1NF dice que para que una tabla la satisfaga, debe ser “isomorfa a alguna relación”. Según el libro “Introducción a los Sistemas de Bases de Datos”, del británico Christopher Date, considerado una verdadera eminenencia en la materia, estas palabras quieren decir que una tabla en 1NF debe cumplir con cinco reglas:



- No debe requerirse un orden entre los registros.
- No debe requerirse un orden entre las columnas.
- No pueden existir registros duplicados.
- Todas las intersecciones de registro y columna deben contener un valor (y sólo uno) del dominio modelado. Aunque Date no lo especifica, la mayoría de los expertos en nuestros días considera como única excepción a esta regla la posibilidad de almacenar un valor nulo para las columnas que no sean clave primaria (recordemos que por “valor nulo” entendemos indistintamente “campo vacío” o “no aplica”).
- Todas las columnas deben ser regulares, en otras palabras, contener sólo valores de los tipos de datos estándar: numérico, alfanumérico y fecha. Una misma columna no puede contener valores de distintos tipos de datos.

Las primeras tres reglas suenan familiares, ya que forman parte de las

definiciones de Edgar Codd respecto de cualquier tabla contenida en un modelo relacional. Las primeras dos son explicativas en sí mismas. Con la tercera regla estamos diciendo “toda tabla debe poseer una clave primaria que permita individualizar unívocamente a cada uno de sus registros”. Pero veamos a qué nos referimos exactamente con el término “clave primaria”.

LOS DISTINTOS TIPOS DE CLAVES

Como ya sabemos, una condición indispensable para que una tabla esté normalizada es que cada uno de los registros contenidos en ella pueda ser identificado en forma unívoca. Para ello, es menester que uno de sus atributos (campos) sea elegido como “clave”, la cual será la encargada de otorgar la unicidad a esa fila de la tabla. Sin embargo, al leer bibliografía sobre esta temática, observaremos que se mencionan distintos tipos de claves, lo cual suele provocar confu-

sión en quienes recién se inician en el terreno de las bases de datos. Aquí detallamos los tipos de clave más comunes y su significado:

CLAVE CANDIDATA: Con este nombre referimos a todos aquellos atributos que son únicos y por ende pueden cumplir la función de identificar un registro unívocamente. Por ejemplo, en una tabla que contenga los empleados de una empresa, probablemente tengamos un campo que almacene el número de legajo, el cual es único para cada empleado, pero también otro que contenga el documento de identidad. Cualquiera de los dos sería adecuado a los fines de individualizar un registro en dicha tabla, pudiendo realizar sobre el mismo operaciones que afecten sólo a ese registro y a ningún otro. Se aplica este término casi exclusivamente a las claves “naturales”.

CLAVE PRIMARIA: Se llama así a el o los campos que son finalmente

TABLA SIN NORMALIZAR	
JEFE	SUBORDINADOS
JORGE	CLARA, ESTEBAN, PEDRO
DIEGO	INÉS, EMILIO

TABLA EN 1NF	
JEFE	SUBORDINADO
JORGE	CLARA
JORGE	ESTEBAN
JORGE	PEDRO
DIEGO	INÉS
DIEGO	EMILIO

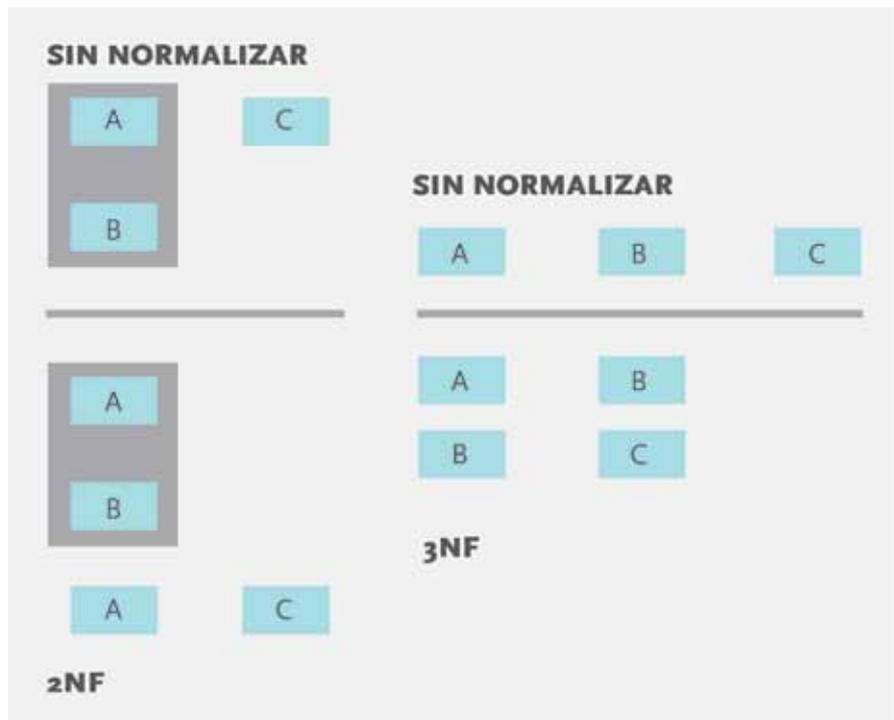
“ Para que una tabla esté en 2NF, debe cumplir con las reglas de la 1NF y además satisfacer la condición de que cualquier campo que no sea parte de la clave candidata dependa de la totalidad de los campos que componen la clave.”

designados para individualizar el registro. Pueden ser “naturales” o “artificiales”.

CLAVE NATURAL: Refiere a atributos ya existentes en la entidad a modelar, que son adecuados para identificar unívocamente a un registro, como los números de legajo o documento ya mencionados. Se dividen en “simples” (compuesta por un único atributo) o “compuestas” (conformada por varios campos, por ejemplo: “tipo de documento” y “número de documento”).

CLAVE ARTIFICIAL: Entendemos por clave artificial a un campo que contiene un valor que no se desprende de los atributos reales de la entidad a modelar, sino que ha sido creado específicamente con el objetivo de individualizar un registro. Por lo general, suele ser una secuencia de números únicos y consecutivos. En la mayoría de los casos se considera la solución más adecuada para generar la clave primaria de una tabla, en especial ante la ausencia de claves candidatas.

LA CUARTA REGLA DE LA 1NF
Si bien las primeras tres reglas de



la 1NF son bastante evidentes, la cuarta es más abstracta y algo más difícil de interpretar, pero puede entenderse a la perfección viéndola en funcionamiento con un caso práctico. Consideremos el siguiente ejemplo.

Dada una tabla “empleado”, con tres atributos: “legajo”, “apellido y nombres”, y “teléfono” pueden

darse los siguientes registros:

- 1, “Fernández, Carlos”, “4000-0001”
- 2, “Echeverría, Claudia”, “3939-6543”
- 3, “Rossi, Ernesto”, “3100-3201”

Ahora supongamos que el empleado con legajo 3 también posee teléfono celular y deseamos almacenarlo en la base de datos. Una

posibilidad sería crear un cuarto registro, repitiendo los valores de los campos “legajo” y “apellido y nombres” de dicho empleado, almacenando el teléfono celular en el campo “teléfono”. Esto no es posible, ya que violaría la tercera regla de la 1NF. Para la entidad (tabla) “empleado”, el campo “legajo” es clave candidata y con seguridad será elegida como clave primaria, por lo cual no podemos tener dos registros con el mismo valor de legajo. La solución más obvia sería almacenar los datos de este empleado en un único registro de esta manera:

3, “Rossi, Ernesto”, “Casa: 3100-3201, Móvil: 15-5000-1138”

Aquí es donde cobra sentido la cuarta regla de la 1NF. Esta solución sería violatoria de dicha regla, ya que en el campo “teléfono” se está guardando más de un valor del dominio correspondiente. Además, para el caso del dominio “teléfono” podría definirse con las características “campo alfanumérico con un máximo de 15 caracteres de extensión”. Con esa restricción, sería directamente imposible almacenar dos o más teléfonos en este campo. Otro enfoque que resolvería este problema, sería tener dos atributos para el dato “teléfono”, por ejemplo “teléfono particular” y “teléfono móvil”. Aunque es una solución harto común entre los programadores de bases de datos, también tiene importantes limitaciones. La que probablemente quede expuesta más rápidamente es que en algún momento se querrá almacenar un tercer teléfono para un empleado; en este caso, nuevamente el programador se verá forzado a crear un tercer campo para este dato, y esto puede llegar a repetirse N veces. Si un empleado tuviera dos teléfonos

móviles, estaríamos creando dos atributos distintos para un mismo dominio de la entidad. Además, tener los distintos teléfonos del empleado separados en varios campos de un mismo registro dificulta enormemente la realización de ciertas consultas. Sin ir más lejos, si quisiéramos obtener el nombre y apellido del empleado que tiene determinado teléfono, deberíamos repetir la consulta tantas veces como distintos atributos tengamos para ese dato. También sería más complejo -y costoso en recursos para la base de datos- determinar si varios empleados comparten el mismo número telefónico. Un diseño acorde a la 1NF implica dividir en dos tablas la entidad “empleado”. Una tabla tendrá los campos “legajo”, y “apellido y nombres”, y otra los atributos “legajo”, “tipo de teléfono” y “número de teléfono”. De esta forma obtendríamos estos registros para nuestro problemático empleado:

Tabla “empleado”
3, “Rossi, Ernesto”

Tabla “empleado-teléfono”
3, “Particular”, “3100-3201”
3, “Móvil”, “15-5000-1138”
3, “Móvil”, “15-4816-2342”

SEGUNDA FORMA NORMAL

Para que una tabla esté en 2NF, debe cumplir con las reglas de la 1NF y además satisfacer la condición de que cualquier campo que no sea parte de la clave candidata dependa de la totalidad de los campos que componen la clave. De esto se deduce que aquellas tablas que estén en 1NF y cuyas claves candidatas sean simples ya de por sí cumplirán con la 2NF. Por lo tanto, deben analizarse cuidadosamente las tablas que tengan claves candi-

datas compuestas para verificar si esta regla se verifica. También por ello hablamos de “clave candidata” y no de clave primaria, ya que una clave primaria “artificial” casi invariablemente estará conformada por un único atributo y por lo tanto no estará expuesta a no cumplir la 2NF. Veámoslo en la práctica. Supongamos que una automotriz produce vehículos en dos de sus plantas. En una tabla “planta-producto” se almacenan los modelos fabricados en cada planta. Dicha tabla cuenta con una clave primaria compuesta por los campos “planta” y “modelo”, y cuenta con un tercer atributo, indicando el tipo de vehículo al que corresponde el modelo:

“Rosario”, “Elegant”, “Sedán”
“Rosario”, “Sportia”, “Cupé”
“Córdoba”, “Elegant”, “Sedán”
“Córdoba”, “Campera”, “Pickup”

Esta tabla no cumple con la 2NF, ya que el tipo de vehículo no depende de la planta en la que se lo fabrique, sino exclusivamente del modelo del automóvil. Puede verse fácilmente que con este diseño se está produciendo una redundancia de datos, ya que en todos los registros pertenecientes al producto “Elegant” se está repitiendo el dato “Sedán”. En este ejemplo, la tabla cuenta con apenas cuatro registros y la redundancia de un solo dato no parece un problema importante. Sin embargo, cuando este problema se produce en tablas con miles o millones de registros, se están desaprovechando valiosos recursos y alargando innecesariamente los tiempos de respuesta. Esto también se soluciona desdoblado la tabla en dos:

Tabla “planta-producto”
“Rosario”, “Elegant”
“Rosario”, “Sportia”



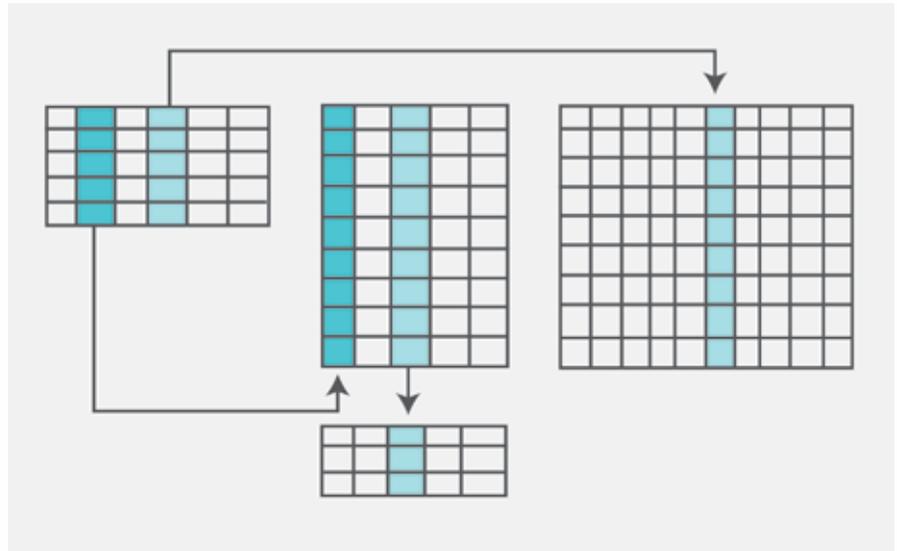
“Entendemos por clave foránea a uno o más atributos, que en una tabla “hija” no son parte de la clave candidata, pero hacen referencia a un registro único en una tabla “padre” en la cual el o los atributos conforman la clave primaria.”

“Córdoba”, “Elegant”
“Córdoba”, “Campera”

Tabla “producto”

“Elegant”, “Sedán”
“Sportia”, “Cupé”
“Campera”, “Pickup”

Además de eliminar las repeticiones, facilita la posterior modificación de datos. Si la automotriz decidiera recatalogar al Elegant como un vehículo “Familiar” en lugar de un “Sedán”, con el diseño que no cumple la 2NF deben actualizarse múltiples registros (todos los correspondientes a las plantas donde



se fabrique el Elegant). Suponiendo que el programador encargado de esta tarea olvide actualizar el registro de una de las plantas, se obtendrán registros inconsistentes: en una de ellas el mismo auto figurará como “Sedán”, y en la otra como “Familiar”. Con el modelo 2NF bastará con actualizar un único registro en la tabla “producto”, eliminando el riesgo de inconsistencias.

TERCERA FORMA NORMAL

Como mencionamos al principio, para que una tabla cumpla con la 3NF, debe satisfacer los requisitos de las formas normales inferiores, es decir, debe respetar la 1NF y la 2NF, y además debe darse que todos los atributos que no pertenecen a la clave candidata deben depender exclusivamente de ésta, y jamás de otros atributos que no conformen la clave. Volvamos a la tabla “empleado” del primer ejemplo y supongamos que le agregamos tres atributos para guardar el domicilio particular de cada empleado: “dirección”, “ciudad” y “código postal”. Los datos podrían quedar así:

1, “Fernández, Carlos”, “Av. de los Perales 2032”, “Buenas Peras”, 2000
2, “Echeverría, Claudia”, “3939-

6543”, “Corralitos 53”, “Buenas Peras”, 2000
3, “Rossi, Ernesto”, “3100-3201”, “San Angel Amadeo 1955”, “Los Parrales”, 2100

Puede deducirse que el atributo “código postal” depende claramente del atributo “ciudad”, el cual no es parte de la clave de la tabla. Los problemas que puede generar esta no-dependencia son similares a los causados por una tabla que no esté en 2NF, pero aquí son más serios, ya que ninguno de los campos afectados pertenece a la clave. Esto posibilita la redundancia del par de atributos “ciudad” y “código postal”, además de complicar la actualización del código postal perteneciente a cada empleado, si la ciudad en la que reside llegara a cambiar de CP. Se soluciona mediante la creación de una tabla “ciudades”, y ambas tablas se relacionarán a través del atributo “ciudad”:

Tabla “empleado”

1, “Fernández, Carlos”, “Av. de los Perales 2032”, “Buenas Peras”
2, “Echeverría, Claudia”, “3939-6543”, “Corralitos 53”, “Buenas Peras”
3, “Rossi, Ernesto”, “3100-3201”, “San Angel Amadeo 1955”, “Los Parrales”

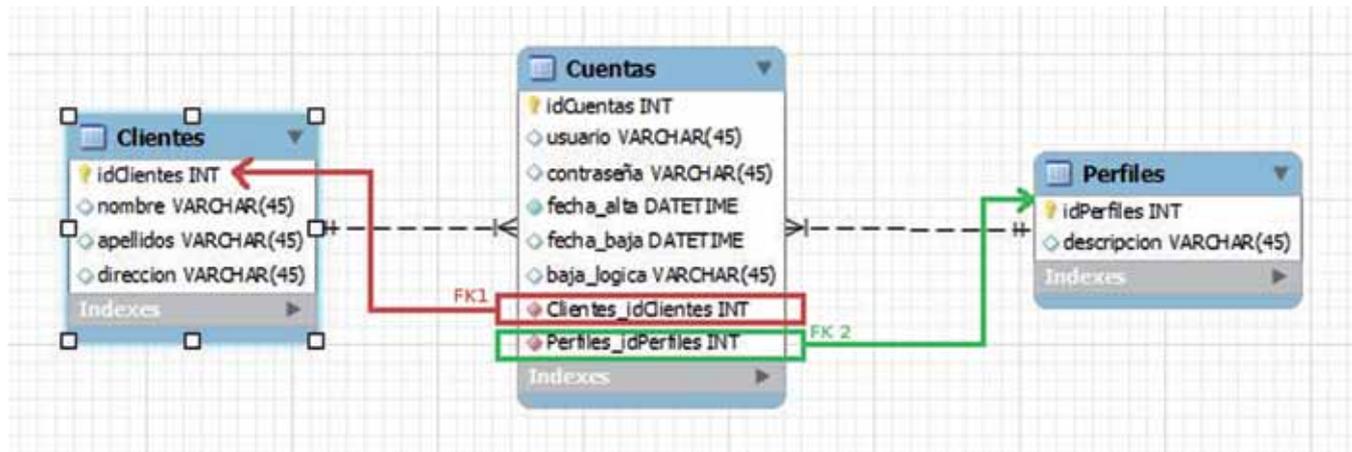


Tabla “ciudad”

“Buenas Peras”, 2000
 “Los Parrales”, 2100

Mientras que el atributo “ciudad” es la clave primaria de la tabla “ciudad”, decimos que en la tabla “empleado” es una “clave foránea”.

CLAVES FORÁNEAS

Las claves foráneas son críticas para mantener la integridad de las relaciones entre entidades. Entendemos por clave foránea a uno o más atributos, que en una tabla “hija” no son parte de la clave candidata, pero hacen referencia a un registro único en una tabla “padre” en la cual el o los atributos conforman la clave primaria. Este tipo de claves son especialmente importantes cuando se opta por el empleo de claves primarias artificiales. Volviendo al ejemplo de la automotriz, supongamos que en la tabla “producto” en lugar de emplear como clave primaria la clave candidata “modelo de vehículo” se utiliza una clave artificial, a la que denominaremos identificador (ID) del modelo:

- 1, “Elegant”, “Sedán”
- 2, “Sportia”, “Cupé”
- 3, “Campera”, “Pickup”

En la tabla “planta-producto” hacemos referencia a cada modelo por su ID:

- “Rosario”, 1
- “Rosario”, 2
- “Córdoba”, 1
- “Córdoba”, 3

Podemos apreciar que el empleo de claves artificiales numéricas también ahorra recursos (memoria y espacio en disco) ya que con apenas un byte podemos tener hasta 256 identificadores únicos, mientras que de emplear como clave primaria la clave candidata “nombre del modelo” requerimos de varios bytes para hacer referencia a los distintos modelos. Para mostrar en un listado los modelos que se fabrican en cada planta, exhibiendo su nombre en lugar de su ID, bastará con programar una consulta en la que se relacionen ambas tablas. Ahora supongamos que por error se elimina el registro con ID de valor 3 de la tabla “producto”. Quedamos imposibilitados de mostrar el nombre del modelo al consultar la tabla “planta-producto”, ya que el registro en la tabla “padre” que contenía dicho nombre ha desaparecido. Si se define “ID del modelo” como clave foránea en la

tabla hija “planta-producto”, haciendo referencia a la clave primaria de la tabla padre “producto”, no podrán eliminarse registros de la tabla padre mientras existan “dependencias” en la tabla hija, garantizando de esta manera la integridad de la información distribuida en más de una tabla al aplicar normalización.

HAY QUE SABER NORMALIZAR, PARA DESPUÉS DES-NORMALIZAR

Nos despedimos con esta frase que solía repetir un sabio profesor universitario. A lo que se refería es que a veces la normalización conspira contra los resultados prácticos de un diseño en particular de la base de datos. Esto es cierto sobre todo a la hora de agilizar los tiempos de respuesta en bases relacionales que manejan enormes volúmenes de información (millones de registros), por lo cual el motor pasa mucho tiempo calculando las relaciones para devolver la información solicitada. Por el momento nos conformamos con enunciar el concepto, ya que profundizar en este sentido sería entrar en terrenos demasiado complejos cuando todavía nos queda mucho camino por recorrer. Seguiremos avanzando en nuestro próximo encuentro. Hasta entonces. **2**



Geolocalización en aplicaciones móviles

Natalia Cantero
natalia.cantero@pixelscode.com

Conceptos generales e implementación en iOS

QUÉ VAMOS A APRENDER:

- qué es geolocalización
- beneficios de integrarla en nuestros negocios y aplicaciones móviles
- productos en el mercado
- introducción a las herramientas que provee iOS SDK

Bienvenidos a esta nueva sección en donde conocerán características innovadoras que podrán

incluir en sus aplicaciones móviles.

El objetivo es que incorporen conceptos y puedan trasladar las nuevas ideas a sus productos, ya sea se encuentren en el ámbito del desarrollo de aplicaciones, o estén buscando cómo proyectar su negocio con la ayuda de los avances en el área de la tecnología móvil.

Hay muchísimos tutoriales y sitios web para que complementen su aprendizaje. A fines de facilitar sus búsquedas, vamos a incluir al pie de

cada segmento palabras claves para que vayan directamente a lo más básico e importante.

QUÉ ES GEOLOCALIZACIÓN

El conocimiento de la ubicación actual del usuario móvil nos permite contextualizar nuestra aplicación ofreciendo contenidos, publicidad y servicios que realmente sean de utilidad, teniendo en cuenta la exigencia de los usuarios actuales dispuestos a esperar poco tiempo a cambio de mucha precisión.

Más allá de qué plataforma móvil elijamos, deberíamos analizar en qué puntos podemos tomar ventaja de la geolocalización para ofrecer, por ejemplo, información relevante al lugar en el cual se encuentra la persona.

Empresas que operan en un mercado global como entidades financieras, cadenas de distribución, redes de agencias, grandes comunidades online, empresas de turismo, entre otras, pueden beneficiarse mucho de estas técnicas en varios aspectos: tener información más precisa de sus clientes para dar soporte a la toma de decisiones, direccionar campañas publicitarias eficazmente o conectando miembros de una comunidad global.

Actualmente, algunos habrán escuchado hablar de aplicaciones de realidad aumentada, la cual combina la visión de un entorno físico del mundo real con elementos virtuales, para dar al usuario una visión enriquecida (ya sea con publicidad, guías urbanas, sugerencias, etc.). En sucesivos artículos nos introduciremos en esta rama de la tecnología móvil, pero es para que sepamos que allí también participa la geolocalización.

[claves] geocoding - geomarketing - augmented reality

APLICACIONES RELACIONADAS EN APPLE STORE

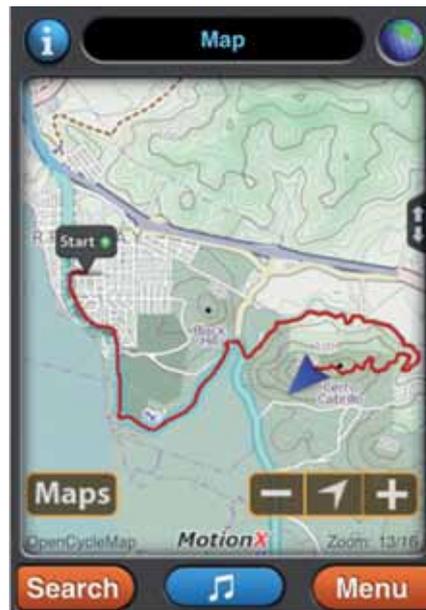
Argentinean Wines

Esta aplicación es un catálogo de las mejores bodegas y vinos del país que permite, además, registrar notas de cata. Apunta a un mercado global e incorpo-

ra la geolocalización para guiar al turismo en su recorrido por los viñedos.

MotionX

A pesar de su precio, esta aplicación encabeza las listas de descarga, ya que permite integrar Facebook en la aplicación también. Integra GPS



para dar soporte a los usuarios que practican pedestrismo y otras actividades al aire libre.

Layar Reality Browser

Es una plataforma móvil que permite ampliar con información el mundo que nos rodea.

Utiliza la tecnología de Realidad Aumentada (AR) para agregar a nuestra vista información sobre turismo, guías y publicidades.

TÉCNICAS DE LOCALIZACIÓN

Existen distintas técnicas de localización geográfica y varían de acuerdo a las características del dispositivo y a las aplicaciones "cliente" que estemos usando.

Cada una de estas tecnologías devuelve información con distinto grado de exactitud, aunque todas tienen cierto margen de error.

En general asociamos la geolocalización con aplicaciones que necesitan saber en qué ciudad, país o



dirección se encuentra la persona, para lo cual GPS cubre las necesidades. Pero en la actualidad existen servicios que nos dan soporte para localizar a un usuario dentro de una oficina, edificio o centro comercial, por ejemplo.

En relación a esto, la empresa Skyhook Wireless ofrece la tecnología que utilizan los iPhone, teléfonos Motorola y otros para descubrir la ubicación de un usuario según las señales WiFi que estén dentro de su alcance. Así mismo, complementa esta búsqueda con los resultados arrojados por torres de celulares y satélites GPS para ampliar la cobertura.

Esta opción, entonces, es recomendable para áreas urbanas e interio-



“El conocimiento de la ubicación actual del usuario móvil nos permite contextualizar nuestra aplicación ofreciendo contenidos, publicidad y servicios que realmente sean de utilidad, teniendo en cuenta la exigencia de los usuarios actuales dispuestos a esperar poco tiempo a cambio de mucha precisión.”

res, en donde la exactitud puede alcanzar los 10 a 20 metros.

COMENCEMOS...

Las herramientas y técnicas que vamos a conocer son para trabajar en aplicaciones nativas de iOS, que son aquellas que se instalan en el propio dispositivo y se desarrollan utilizando el SDK de Apple para aplicaciones móviles.

Es por esto que cuando compartamos código fuente en un futuro, será en lenguaje Objective-C.

Es importante aclarar, para aquellos programadores de tecnologías web, que pueden incorporar geolocalización sin necesidad de que sus productos se instalen en celulares con GPS. Pueden investigar cómo Google Maps da soporte a este escenario sabiendo que van a tener algunas limitaciones dadas por la propia tecnología.

En este primer artículo vamos a obviar incluir código fuente, porque sólo vamos a lograr confundirlos con tanta información. Pensamos que es más útil que entiendan los conceptos y puedan seguirlos ustedes mismos desde las fuentes que publica Apple.

- Si aún no son desarrolladores iOS, tienen que comenzar por instalar el SDK en una computadora que corra el sistema operativo MAC OS X Snow Leopard.

- Tanto para descargar el SDK como para ver nuestro código de ejemplo, tienen que acceder a:
<http://developer.apple.com/iphone/>

- Crean una cuenta de usuario sin costo y acceden al sitio oficial de Apple, que cuenta con todo lo que necesitan.

- El proyecto que deben abrir con Xcode se llama LocateMe. Allí van a ver implementado todo lo que vamos a compartir a continuación.

CORE LOCATION

Es la librería (framework) disponible en el SDK que, haciendo uso del hardware disponible en el equipo, nos permite conocer la ubicación actual del usuario.

A través de sus clases y protocolos, podemos configurar cómo será la recuperación de las coordenadas y el comportamiento de nuestra aplicación respecto de las mismas.

Las coordenadas generalmente toman la forma de dos valores numéricos, latitud y longitud que representan un punto correspondiente en el mundo.

Si se fijan en el proyecto LocateMe, se incluye en la carpeta Frameworks a CoreLocation.framework. El siguiente paso es crear una instancia del objeto CLLocationManager, el cual será manejado por el delegado CLLocationManagerDelegate.

El comportamiento más básico que necesitan para obtener la localización del usuario podrán configurarlo a través de las propiedades que expone este nuevo objeto que han creado.

La mejor guía que encontramos para que aprendan cómo deben trabajar con CoreLocation y todas las herramientas que veremos a

continuación, está en el sitio: <http://developer.apple.com/videos/iphone/> y se llama Finding Your Way with Location and Maps

Les sugerimos que se tomen unos minutos para verlo, porque van a ahorrar muchísimo tiempo y confusión intentando imaginar cómo interactúan los elementos de este framework que les estamos presentando.

[claves] `CLLocationManager` - `CLLocation` - `CLLocationManagerDelegate`

BUENAS PRÁCTICAS AL USAR CORELOCATION

iPhone devuelve información sobre la ubicación del usuario de maneras variadas, dependiendo de qué red (network) y características del dispositivo (hardware) están disponibles.

Cada uno de estos métodos tiene una exactitud (accuracy) asociada.

- Torres de celulares tienen una exactitud aproximada de 1 a 3 kilómetros
- Redes Wi-Fi con una exactitud cerca de 100 metros
- GPS provee una exactitud de hasta menos de 40 metros

En la medida que nos acercamos a la mejor exactitud posible, también incrementamos el tiempo necesario para obtener información de la ubicación. Y la energía que toma devolver estos datos, también aumenta.

Es por esto que debemos siempre recordar especificar la exactitud requerida para nuestra aplicación. Por ejemplo, si pretendemos obtener el clima de una ciudad, probablemente no necesitemos fijar la precisión en



10 metros. En cambio, con 3 kilómetros puede ser suficiente.

- Chequear la configuración global del iPhone antes de comenzar a pedir actualizaciones de la ubicación. Los usuarios de iPhone pueden habilitar o restringir el acceso de las aplicaciones al servicio de Localización, por lo cual es recomendable verificarlo antes de avanzar, para no caer en errores no manejados.

- Conservar la energía del dispositivo usando temporizadores (timers) Cuando intentamos obtener una localización hay tres tipos de errores que podemos recibir:

`kCLErrorDenied`: acceso denegado por el usuario
`kCLErrorNetwork`: red no disponible o error en la red
`kCLErrorLocationUnknown`: imposibilitado de obtener una localización. Este es un tipo de error temporal, que puede estar dado porque el usuario está en un ambiente que dificulta el posicionamiento (atravesando un túnel, en el subterráneo, etc).

En estos casos, uno puede intentar nuevamente obtener una localización luego de unos instantes, pero

no deseamos intentar para siempre, ya que esperar el resultado más preciso consumirá demasiada batería del equipo.

[claves] `CLLocationManager` - `desiredAccuracy` - `timeStamp`

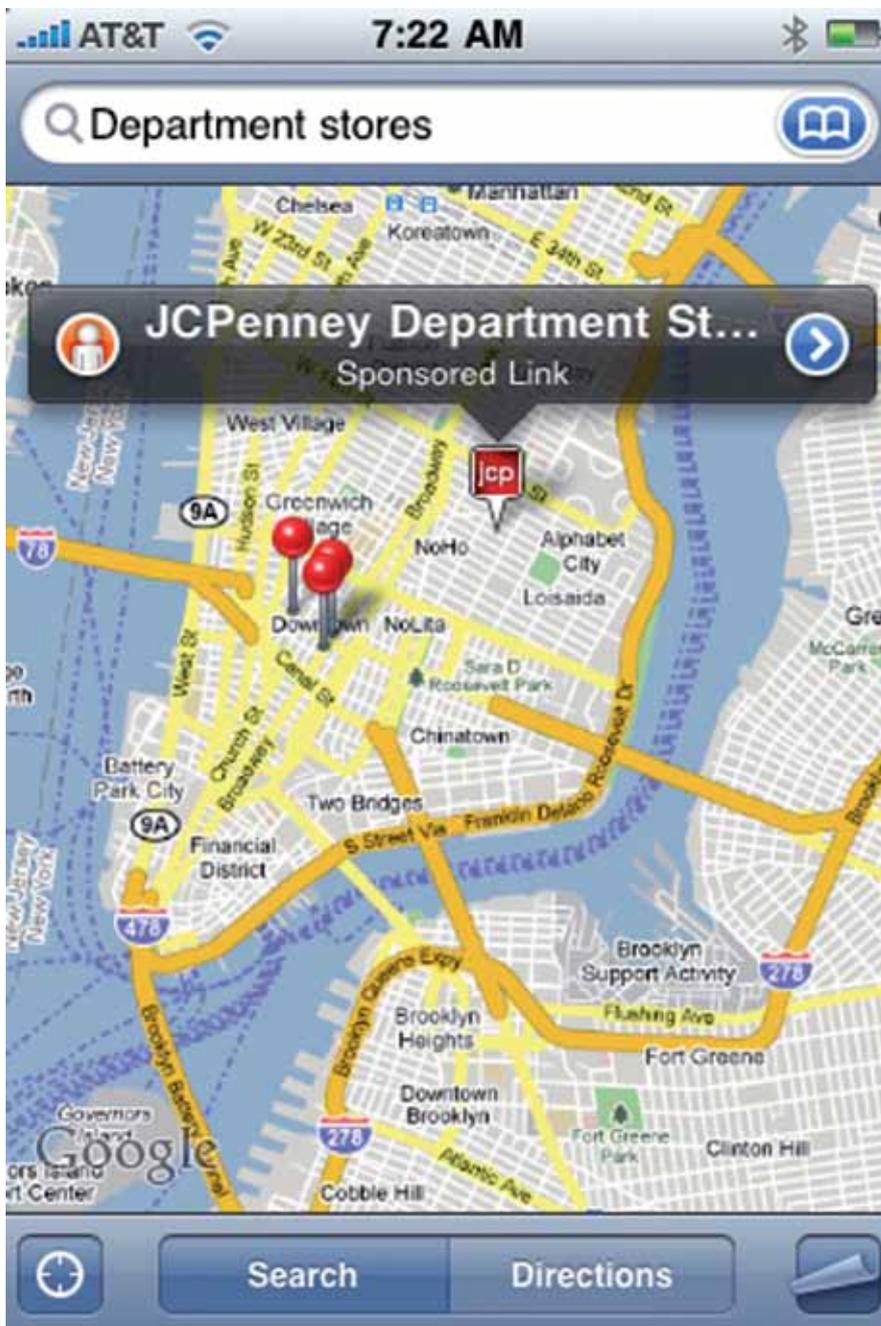
MAP KIT

MapKit es el framework incluido a partir de iPhone OS 3.0, que nos provee de todas las herramientas que necesitamos para incluir en nuestra aplicación un mapa con la localización que obtuvimos previamente usando CoreLocation.

También nos da soporte para incluir anotaciones en el mapa y realizar búsquedas de geocodificación inversa para marcar la localización actual del usuario.

Para los que se estén iniciando, este último paso es necesario para traducir una localización con el formato latitud, longitud (x, y) a una dirección válida y entendible para el usuario (calle, número, ciudad, etc).

Esta integración se realiza trabajando con una vista del tipo `MKMapView`



View que incluye las características de ampliar el mapa a través del evento multi-touch y elegir qué tipo de mapa vamos a incluir (regular, satelital, híbrido), entre otras.

También usamos la propiedad region del MKMapView para posicionarnos en un sector específico

del mapa y con un determinado zoom, a fines de optimizar la vista total acorde a la región que queremos mostrar.

Para finalizar, es importante aclarar que MapKit trabaja con el servicio de Google Maps para móviles, de modo que sugerimos

que para incluirlo en su aplicación para iPhone se familiaricen con las condiciones de este servicio (<http://code.google.com/apis/maps/iphone/terms.html>)

[claves] MapKit - MKMapView – MKAnnotationView

ANOTACIONES

Una vez que tenemos nuestro mapa publicado y corriendo en la aplicación, el próximo paso que podemos desear es decorar nuestro mapa con anotaciones (annotations). Entre los tipos de anotaciones que podemos incluir encontramos el clásico pin rojo o nuestro propio marcador.

Las anotaciones siempre se refieren a una coordenada exacta del mapa.

La vista que da soporte a esta funcionalidad se llama MKAnnotationView y van a encontrar en las referencias qué funciones pueden usar para enriquecer su comportamiento y visualización.

[claves] MKAnnotationView

GEOCODIFICACIÓN INVERSA

Antes comentamos que necesitábamos de la geocodificación para poder procesar coordenadas geográficas (latitud y longitud), para asignarlas a puntos específicos de un mapa.

Como el framework CoreLocation no nos devuelve direcciones, sino



“ Las herramientas y técnicas que vamos a conocer son para trabajar en aplicaciones nativas de iOS, que son aquellas que se instalan en el propio dispositivo y se desarrollan utilizando el SDK de Apple para aplicaciones móviles.”

coordenadas, es fundamental que nos apoyemos en la geocodificación inversa.

Estas coordenadas geográficas pueden usarse para localizar el punto del mapa en el Sistema de Información Geográfica (SIG) o para ser embebidas en medios de comunicación tales como fotos digitales a través del etiquetado geográfico (geotagging).

RESUMIENDO

Forward Geocoding es el proceso por el cual transformamos una dirección en una coordenada y Reverse Geocoding toma una coordenada y devuelve su dirección relacionada

MapKit solamente provee reverse geocoding, pero si estamos interesados en el proceso inverso, podremos hacer uso de cualquier herramienta de las que existen en línea.

[claves] Forward Geocoding - Reverse Geocoding – MKReverseGeocoder – MKReverseGeocoderDelegate - MKPlacemark



CONCLUSIÓN

Nuestro objetivo fue resumir en palabras entendibles lo que realmente es básico, sabiendo que los lectores vienen de diferentes niveles de conocimientos previos, áreas y negocios.

Estamos seguros de que en las próximas ediciones podremos ir adaptando los contenidos para que realmente los ayuden y alienten en este camino, por eso contamos con sus comentarios y sugerencias.

Nuestra sugerencia es poner manos a la obra... practicar, investigar, preguntar y jugar con las herramientas porque, en la medida que se familiaricen, van a querer enriquecer sus aplicaciones.

No importa que ya existan muchos productos móviles en las tiendas online, siempre podemos marcar una diferencia con la calidad de nuestro trabajo y con la creatividad con la cual resolvamos lo que otros ya han hecho. **P**

Frameworks en PHP: CodeIgniter, una gran opción

Cristian y Gabriel Gaitano Ornia

cristian.gaitanoornia@pixelscode.com

@cerge74



Cuando se presentó la posibilidad de escribir un artículo sobre Frameworks en PHP, la decisión se tornó un tanto compleja, puesto que más allá del interesante desafío que eso implica, hoy en día existen tantos entornos como entusiastas tiene cada una de esas 'mega' librerías.

La elección por CodeIgniter esencialmente surgió porque considero es una opción a la altura de otros frameworks de gran prestigio, pero que a diferencia de ellos ofrece un grupo de librerías más accesibles en cuanto a su instalación y uso. Asimismo, la curva de aprendizaje (que modestamente trataremos de iniciar en este artículo), es relativamente suave y me parece surge como una excelente opción, incluso para quien no ha profundizado mucho en el patrón MVC. Si es ese el caso, sugerimos poder profundizar un poco más en ese apasionante paradigma, quizás hoy el más usado y divulgado.

De todas formas, empezaremos aclarando un poco en qué consiste MVC, o al menos sus partes integrantes, sin caer en teorías demasiado pesadas.



EL PARADIGMA MVC

MVC está formado por las siglas que representa Modelo-Vista-Controlador o, en inglés, Model-View-Controller. Llegado este punto es bueno poder analizar cada una de ellas, para así poder entender el concepto:

Los Controladores son los encargados de gestionar toda la lógica de la aplicación y para ellos se valen de métodos (o funciones de clase). Vale decir, manejan tanto el flujo de información que procesa el modelo como qué vista será la encargada de hacer visibles esos datos.

El Modelo, esencialmente, procesa todo lo referente a datos y pone el foco en la estructura de los mismos abarcando el grupo de funciones que gestionan la información procedente de la base de datos. Por ejemplo, si tenemos una tabla llamada “clientes”, deberíamos tener un modelo para esa tabla y funciones dentro de ese modelo para seleccionar, crear, actualizar y eliminar registros para esa tabla “clientes”, no importa qué motor de datos usemos.

Las Vistas representan la interfaz de usuario que vemos en la pantalla, siendo en este punto donde formateamos y determinamos cómo se verá la misma.

En nuestro ejemplo más adelante veremos un caso en que usamos tanto MVC, así como otro que sólo utiliza Vista-Controlador.

QUÉ ES CODEIGNITER

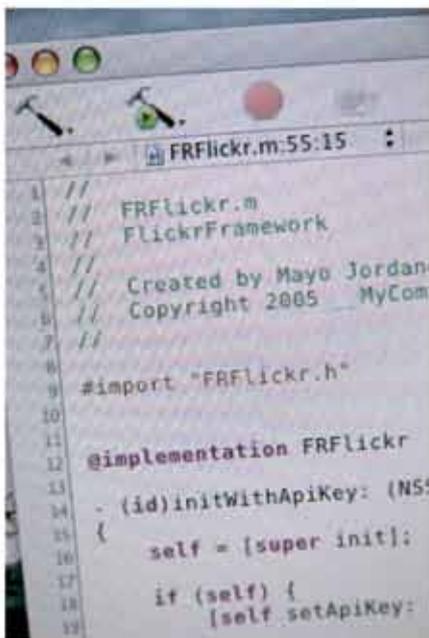
Como posiblemente ya saben, los frameworks nos ofrecen la posibilidad de gestionar librerías que nos ayuden en la rutina diaria de crear formularios, acceder a bases de datos y otras tantas estructuras de código partiendo de una base más segura, eficiente, solida, fácil de usar (y actualizar) y por sobre todo... sin tener que reinventar siempre todo de cero. Sé que muchos programadores tendrán sus propias librerías y todo esto quizás podrá sonar poco atractivo, pero es interesante poder probar estos entornos ya que permiten no sólo integrarnos a más y mejores ofertas de trabajo, sino utilizar y participar de módulos creados y testeados por una extensa comunidad que agrega y mejora características todo el tiempo.

CI es framework Open Source liviano y rápido que ofrece eso y mucho más. Vale decir, posee un acceso a sus librerías bien estructurado y tiene compatibilidad tanto para PHP 4 como para PHP 5, amén de ofrecer extensa documentación y una comunidad que crece día a día.

En general, además, siempre se ha destacado cuando se necesita privilegiar el rendimiento de aplicaciones web que corren en hosting compartidos, así como cuando se ejecutan muchas versiones de PHP con diferentes tipos de configuraciones.

Pero esto no pretende aburrir ni ser un panfleto de propaganda de CI, sino más bien un buen acercamiento para aquel que no trabaja con frameworks en PHP y que desea hacerlo sin pasarse extensas horas configurando o aprendiendo detalles a veces superfluos o poco productivos. Considerando la forma en que crece CI, y eso que decíamos antes sobre su buena documentación y la fácil curva de aprendizaje, creemos que es una buena opción a la hora de lograr esos objetivos iniciales.

“La elección por CodeIgniter esencialmente surgió porque considero es una opción a la altura de otros frameworks de gran prestigio, pero que a diferencia de ellos ofrece un grupo de librerías más accesibles en cuanto a su instalación y uso.”



DESCARGAR E INSTALAR CODEIGNITER

Se puede descargar la última versión desde el sitio <http://codeigniter.com>. En este momento, CI va por la versión 2.0.1 y sobre ella basamos nuestro ejemplo. No querríamos perder tiempo en repetir la guía de usuario de instalación. Además, como ya comenté, una de las razones por las que recomendamos CI es debido a su fácil proceso de instalación, ya que más allá de que podamos configurar detalles es esencialmente descargar y empezar a usar.

De todas formas es interesante analizar opciones y acciones a tener en cuenta. En cuanto a la instalación, obviamente, puede ser en modo local o en el server. En mi caso, lo utilicé en los directorios de publicación, según estuviera trabajando con Appserv, Wamp o Xampp y siempre sin mayores problemas. En el server, usualmente, también se copia en el raíz del sitio pero nada impide que se ejecute dentro de subdirectorios.

Lo que sí es importante verificar siempre es la url base de la aplicación web, que viene a ser la dirección para acceder a la raíz de CodeIgniter, según el servidor y directorio donde hayamos colocado el código fuente del framework.

Esencialmente, todo el trabajo se centra en verificar el archivo de configuración que se encuentra en `application/config/config.php`, revisando la variable de configuración que se carga en `$config['base_url']`.

Dicho archivo tiene muchas opciones por defecto que pueden, directamente, configurar tanto opciones de encriptación como manejo de sesiones y cookies.

Es importante destacar que para el manejo de base de datos deberemos centrarnos en editar el archivo `application/config/database.php`, indicando ahí los usuales parámetros referidos a la conexión con el servidor de base de datos (nombre del servidor, usuario, contraseña y nombre de la base de datos):

```
$db['default']['hostname'] = "localhost";
$db['default']['username'] = "nombre de usuario de MySQL";
$db['default']['password'] = "contraseña de usuario de MySQL";
$db['default']['database'] = "nombre de tabla";
```

Pero no nos adelantemos más, luego del primer proyecto veremos realmente cómo poder emitir un listado de datos en minutos.

INICIÁNDOSE EN LA PRÁCTICA DE MVC Y CODEIGNITER

Más allá de que podamos encontrar ejemplos en la web, lo interesante es que podamos allanar el camino en forma sencilla, y sorteando errores y problemas que suelen presentarse y no suelen estar comentados en los tutoriales de consulta obligados. En ese sentido, nos orientaremos tratando de lograr un acercamiento más ameno y práctico, alejándonos un poco de los usuales “Hola mundo”.

Como vimos, pondremos el foco en tres carpetas especialmente, primero en `/controllers`, luego en `/views` y finalmente en `/models`.

Podremos ver resultados en ese orden, trabajando primero con un controlador que muestre un mensaje por pantalla, luego con vistas que permitan visualizar información desde un controlador y, finalmente, puro MVC con un trabajo conjunto entre el controlador que gestionará la vista y el modelo.

1) Con el controlador solo archivos que usaremos: controllers/test.php

Por empezar, notaremos que cada clase controlador debe ir con mayúscula y derivarse de la clase madre CI_Controller (en muchos manuales verán solo 'Controller', pero el nuevo motor usa esa forma). Es decir que para el archivo controllers/test.php tendremos una clase Test.

```
class Test extends CI_Controller {
```

```
}  
}
```

Asimismo, para acceder a ella deberemos usar la forma:

<http://mipagina/micodeigniter/index.php/controlador/>

... o, por ejemplo, si lo tienen en forma local:

<http://localhost/CodeIgniter/index.php/test/>

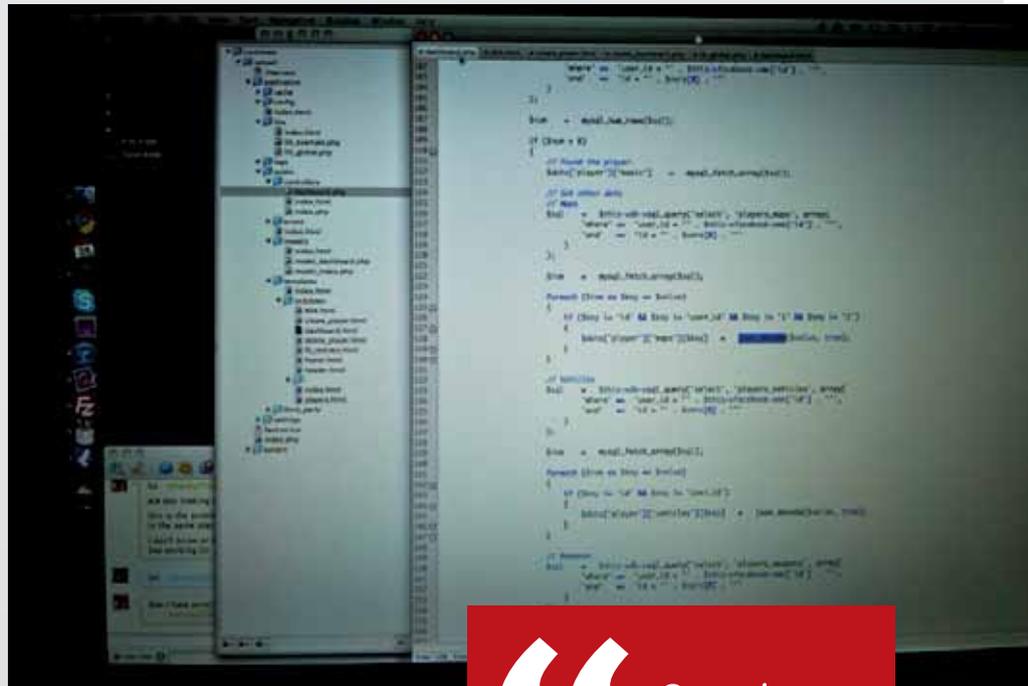
Notar que es importante la barra [/] luego del nombre del controlador.

Ahora, le agregaremos una función con parámetros a esa clase, y luego accederemos a esa función desde la misma url que manejamos antes. Cada función en un controlador de CI representa un destino o ruta. De esa forma, si tenemos una ruta del tipo test/testear, el controlador tendrá una función llamada testear().

En nuestro ejemplo, cargaremos nuestro nombre y edad de esta forma:

```
class Test extends CI_Controller {
```

```
function testear($nombre, $edad)  
{  
    echo "Mi nombre es ".$nombre. "<br>";  
    echo "Mi edad es ".$edad;  
}  
}
```



“ Como vimos, pondremos el foco en tres carpetas especialmente, primero en /controllers, luego en /views y finalmente en /models.”

```
153 mysql_select_db("bookmarkiberia ");
154 $result = mysql_query("SELECT * FROM settings WHERE key = 'maintenance'");
155 $row = mysql_fetch_object($result);
156 if ($row->value == "true")
157     $maintenance = 1;
158 else
159     $maintenance = 0;
160
161 if ($maintenance):
162     include "pages/maintenance.php";
163 else:
164     if (isset($_GET['page'])):
165
166         $page = addslashes($_GET['page']);
167
168         if (file_exists("pages/.$page.".php)):
169             include "pages/.$page.".php";
170         else:
171             include "pages/home.php";
172         endif;
173     else:
174
175
176         include "pages/home.php";
177     endif;
```

“ En dicho archivo referenciamos ese array llamado \$datos, pero, atención, de otra forma. Es decir, sólo por el nombre de elemento que lo diferencia, pero convertido en variable php.”

http:mipagina/micodeigniter/index.php/controlador/funcion/parametro1/parametro2

y cargaremos información accediendo así:

http://localhost/CodeIgniter/index.php/test/testeardatos/cristian/36

El resultado es obvio, aparecía en pantalla:

Mi nombre es Cristian

Mi edad es 36

2) Controlador y vista generando información en pantalla

Archivos que usaremos: controllers/test.php y views/test_view.php

Nuestro siguiente ejemplo utiliza sólo ‘dos patas’ del patrón MVC, el controlador y la vista. De esa forma usaremos arrays con variables cargados desde el controlador y que pasaremos a la vista.

Utilizaremos un array llamado \$datos, que cargue tres nombres:

```
function vc()
{
    $datos['nombrevar1'] = 'Gisela';
    $datos['nombrevar2'] = 'Cristian';
    $datos['nombrevar3'] = 'Juan Ignacio';
}
```

Y luego sí, pasaremos a cargar la vista con la forma que se utiliza en Codeigniter, es decir, utilizando la referencia \$this (el controlador) y los métodos load y view, que cargan el nombre de la vista (pensémoslo como el nombre del archivo, pero sin el “.php”), y pasándole como segundo parámetro el array de nombres llamado \$datos:

```
$this->load->view('test_view', $datos);
```

De esa forma, la clase completa nos quedaría:

```
function vc()
{
    $datos['nombrevar1'] = 'Gisela';
    $datos['nombrevar2'] = 'Cristian';
    $datos['nombrevar3'] = 'Juan Ignacio';
    $this->load->view('test_view', $datos);
}
```

Pasemos entonces a la vista, que está contenida físicamente en views/test_view.php. Esto sería simplemente un archivo con código html, o bien con código php que genere html.

En dicho archivo referenciamos ese array llamado \$datos, pero, atención, de otra forma. Es decir, sólo por el nombre de elemento que lo diferencia, pero convertido en variable php. Es decir que si llamábamos a \$datos['nombrevar3'] = 'Juan Ignacio'; en el modelo, si queremos que esto se vea en la vista llamaremos a \$nombrevar3'. Es importante entender bien esto para después no sufrir errores.

Por ejemplo:

```
<html>
<head>
<title>Probando una vista de Codeigniter </title>
</head>
<body>
<h1><?php echo $nombrevar1 ." y " . $nombrevar2 ." se encuentran con " . $nombrevar3 ?></h1>
</body>
</html>
```

Con esa vista o template si seguimos la estructura antes dada y accedemos a esta url de ejemplo,

<http://localhost/CodeIgniter/index.php/test/vc>

... veremos el mensaje:

Gisela y Cristian se encuentran con Juan Ignacio

3) Puro MVC: Modelo, vista y controlador trabajando juntos

Archivos que usaremos: controllers/test.php, models/test_model.php y views/test_view.php

Llegamos entonces al ejemplo que reúne los 3 archivos. Empezaremos por generar el nuevo integrante del ahora flamante trío, para después modificar puntualmente el controlador y vista que ya tenemos trabajando.

Empecemos por el nuevo integrante, es decir el modelo. Como decíamos al iniciar la nota, y sobre su papel dentro de MVC, el modelo se encarga esencialmente de manejar datos, sean estos provenientes de una base de datos o, como en nuestro ejemplo, que necesitemos simplemente mostrar una variable de texto. De todas formas, imaginen que el modelo podría también formatear datos de fechas o bien realizar un cálculo que modificara dicho valor y lo 'devolviera' al controlador, para que éste lo 'entregue' a una vista.

En CI, un modelo esencialmente se crea de esta forma dentro de la carpeta /models y siguiendo las condiciones generales vistas anteriormente:

```
class test_model extends CI_Model {
}
```

“CI proporciona una clase para validar los formularios de una manera muy sencilla y, por sobre todas las cosas, más segura (previniendo ataques XSS o por inyección SQL).”

Luego, agregaríamos una función llamada `cargaNombre`, que invocaremos en el modelo y que retornará un valor.

```
function cargaNombre($nombreP_Model) {  
    return $nombreP_Model;  
}
```

Por lo cual, el modelo completo sería:

```
class test_model extends CI_Model {  
  
    function cargaNombre($nombreP_Model) {  
        return $nombreP_Model;  
    }  
  
}
```

En el controlador tendríamos una función que lo invoca de esta forma:

```
function mvc()  
{  
  
    $this->load->model('test_model');  
    (...)  
}
```

En nuestro ejemplo, intentaremos que retorne un valor que le pasamos por parámetro desde el controlador (o bien, en términos didácticos, en la url).

Para eso, agregamos primero el parámetro de esta forma:

```
function mvc($nombrep)
```

Para luego cargarlo dentro de un método del modelo llamado `cargaNombre`:

```
$datos['nombrevar'] = $this->test_model->cargaNombre($nombrep);
```

Manteniendo la misma línea de llamada a la vista, nuestro código quedaría así:

```
function mvc($nombrep)  
{  
  
    $this->load->model('test_model');  
    $datos['nombrevar'] = $this->test3_model->  
>cargaNombre($nombrep);  
    $this->load->view('test_view', $datos);  
}
```

La vista quedaría reducida a mostrar ese nombre que venimos manejando en una función. Por eso, podemos actualizar nuestra vista anterior o, simplemente, en el controlador direccionar a una nueva página (ejemplo: `test_view2`):

```
<html>
<head>
<title>Probando una vista de CodeIgniter </title>
</head>
<body>
<h1><?php echo "El nombre es ". $nombrevar ?></h1>
</body>
</html>
```

¿Cómo cargar el parámetro? Simple, en la url escribiríamos algo así:

http://localhost/CodeIgniter/index.php/test/mvc/cristian

MANEJO DE BASES DE DATOS

Anteriormente vimos cómo configurar nuestro framework para que conecte a una base de datos. Una vez realizado eso, procederemos entonces a poner manos a la obra con el objetivo de conectar por primera vez a la base de datos y así mostrar por pantalla un listado de contactos.

Partiremos de una muy pequeña tabla de contactos como esta:

Nombre: contactos

Campos: id (numérico autonumérico), nombre y email (ambos varchar)

O bien con la orden sql:

```
CREATE TABLE contactos (
  id INTEGER AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR 200,
  email VARCHAR 100
);
```

Esta aplicación incluso no será puro MVC, ya que se abstendrá de utilizar un modelo. Esto nos permite ver el poder de CI y centralizar nuestra atención en el controlador y la vista.

Analicemos un poco entonces el código del controlador:

```
<?
//Creamos el controlador como una instancia de la clase CI_Controller
class Contacto extends CI_Controller {
//Generamos la función o método VerListado
function VerListado() {
//Conectamos a la base de datos
$this->load->database();
//Ejecutamos la consulta que trae ambos campos
$query = $this->db->query("SELECT contact_lastName, contact_firstName FROM contact");
//Si hay registros los cargamos en variables
```

```

        if($query->num_rows() >0) {
            foreach($query->result_array() as $row) {
                $contactos[] = $row;
            }
            $data['contactos'] = $contactos;
        }
        // Y le decimos a la vista que los muestre
        $this->load->view('contacto_view', $data);
    }
}
?>

```

Repasemos entonces el código de nuestra vista de datos:

```

<html>
<head>
<title>Probando BD en Codeigniter</title>
</head>

<body>
<table border="1" cellspacing="1" cellpadding="5">
<tr>
<td><b>Apellido</b></td> <td><b>Nombre</b></td>
</tr>
<?

```

//Si tenemos un array con datos de contactos

//Obsérvese que traemos la variable \$contactos de forma similar a lo que vimos en nuestros ejemplos iniciales.

```

if(is_array($contactos)) {
    //Iteramos mostrándolo en una tabla
    foreach($contactos as $key => $valor) {
        ?>
        <tr>
        <td><? echo $valor['apellido']?></td>
        <td><? echo $valor['nombre']?></td>
        </tr>
    }
} else {

```

//O bien avisamos que no hay registros

```

?>
        <tr><td colspan="2">No se encontraron contactos!</td></tr>
    }
    ?>
?>
</body>
</html>

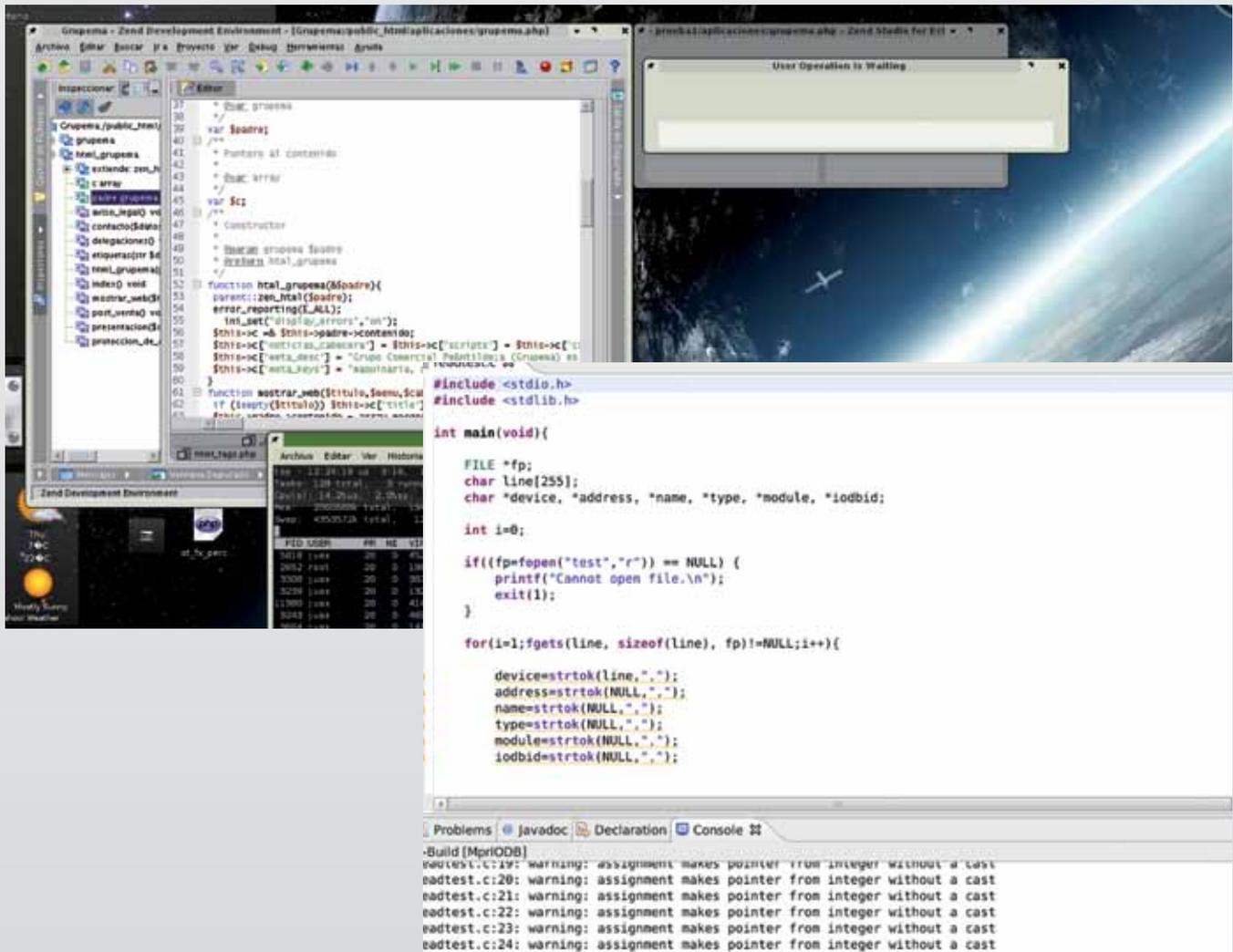
```

Podemos visualizar el listado escribiendo algo así:

<http://localhost/CodeIgniter/index.php/contacto/verlistado>

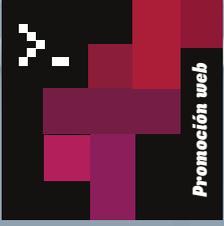
Es interesante destacar que el listado también se mostrará si ponemos:

<http://localhost/CodeIgniter/index.php/contacto/VerListado> (con mayúsculas, como el nombre del método original).



CODEIGNITER, LIBRERÍAS ADICIONALES Y OTRAS TECNOLOGÍAS

Es importante destacar también cómo CodeIgniter ofrece excelentes herramientas de validación de formularios y paginación. Si bien esto da para muchos artículos más, es interesante resaltar que por ejemplo CI proporciona una clase para validar los formularios de una manera muy sencilla y, por sobre todas las cosas, más segura (previniendo ataques XSS o por inyección SQL, como vimos en nuestro artículo de seguridad de Pixels&Code #1). En efecto, con la Validation Class podemos establecer reglas y pautas de comportamientos para verificar y ordenar campos de formularios de una forma ordenada y accesible. Pero no se termina ahí el tema, si necesitamos algo más a medida, CI también nos permite en forma relativamente sencilla crear nuestras propias funciones de validación de datos. Asimismo, CodeIgniter incluye clases nativas que ofrecen la posibilidad de poder paginar datos y manejar tablas, algo muy útil cuando se necesita mostrar una considerable cantidad de información. También se ofrecen excelentes opciones de acoplamiento con tecnologías como Ajax y librerías de JQuery, siendo también factible conectar CI con redes sociales como Facebook y Twitter. En definitiva, CI aparece hoy como una más que interesante opción a la hora de buscar un framework sólido pero también muy accesible de aprender e instalar, tanto para desarrolladores novatos como para los más avanzados. Ojalá este artículo pueda servir como punto de inflexión en el inicio de esa búsqueda tan apasionante. **P**



Promoción web



Aprovechando Las redes social es como herramientas de difusión

Adrián Ortiz
adrian.ortiz@pixelscode.com
@DemasiadoHumano

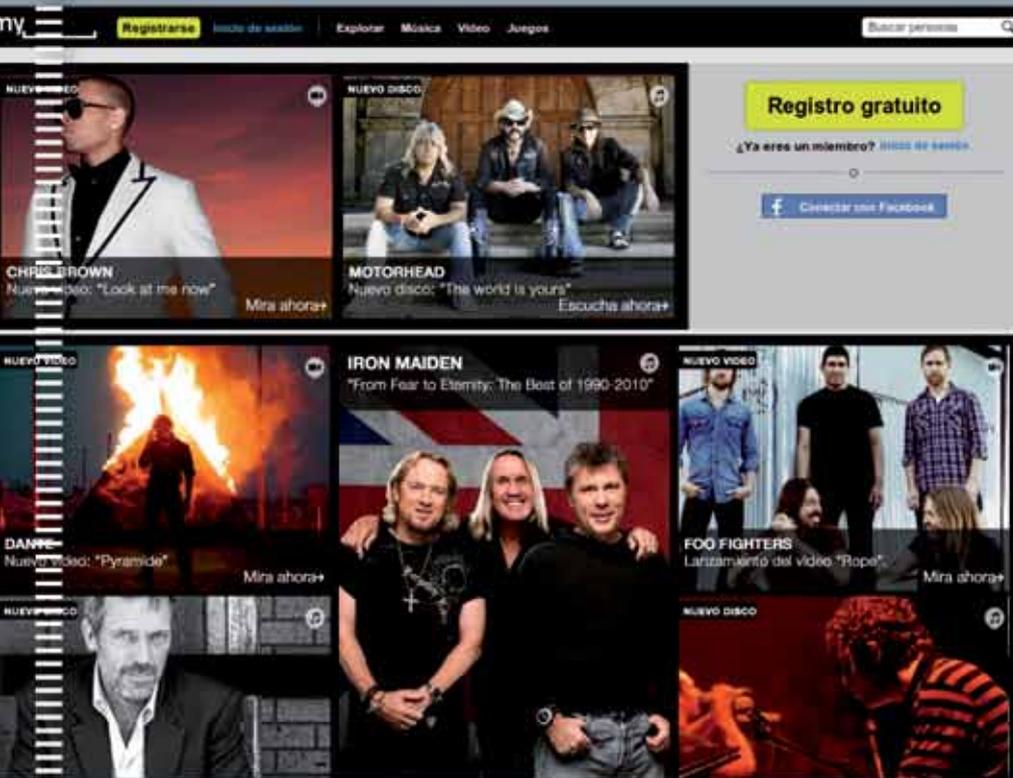


Las redes sociales constituyen actualmente uno de los focos que mayor atención reciben dentro del mundo virtual que es Internet. Y si bien existen muchos puntos poco claros en el uso que se le pueden dar a la hora de difundir contenido, hoy son una herramienta ineludible si se trata de planificar la comunicación de un emprendimiento web.

En los últimos meses de 2010 surgió con muchísima fuerza el concepto de social media o community management para aglutinar e intentar categorizar muchas de las teorías e ideas que giraban en torno al uso de las redes sociales virtuales como herramientas de difusión de contenidos, como también para planear y llevar a cabo una estrategia de marketing online apropiada. Dentro de este campo, se busca tener en cuenta tanto a la marca, producto o servicio en sí, como así también a los usuarios finales y cómo la información que estos proporcionan en forma de retroalimentación puede servir

para pulir y retocar tales estrategias. Asimismo, conceptos como SEO (search engine optimization) y SEM (search engine marketing), que ya hemos tratado en una nota previa, son temáticas muy cercanas a todo lo que es social media, complementarias con los conocimientos que se encuentran en desarrollo dentro del campo que comprende lo que es social media. Y cabe mencionar a lo que se llama SMO (social media optimization), como el núcleo más duro de lo que serían las campañas y estrategias desarrolladas en redes sociales cuyas finalidades más notorias son netamente publicitarias o comerciales.

¿Qué sería o en qué consiste, concretamente, lo que llamamos social media o community management? Básicamente, y al momento actual, es un conjunto de conceptos cuyo foco reside en la relevancia que poseen las redes sociales como lugar de participación de los usuarios, para crear o intercambiar contenido u otro tipo de experiencia. Está íntimamente ligado a la idea de Web 2.0, pero es una profundización de la misma, ya que expande el campo de acción de los usuarios, y los pone en el mismo nivel de quienes emiten el mensaje, sean empresas, medios de comunicación o estrellas del mundo del espectáculo o el deporte. En cierto punto, se cree que las redes sociales, junto con otros servicios como YouTube, son actores estelares de lo que se da en llamar Web 3.0, nueva etapa de la evolución social de Internet, en la cual el usuario es el protagonista principal. Otro de los puntos a



El otrora sitio pionero de las redes sociales, y el primero en alcanzar la masividad, hoy se encuentra en un proceso de reestructuración cuyo perfil apunta a la industria musical.

destacar de ambos conceptos es su utilización de la idea de viralidad, de cómo un comentario, negativo o positivo, puede pasar de una persona a la otra, de manera continua. El clásico “de boca en boca”.

También aquí podemos incluir un concepto del mundo de los negocios, muy expandido también, que es el e-commerce, comercio electrónico, que comprende a toda compra y venta de productos o servicios por Internet, y a los sistemas que hacen esto posible. Asimismo, todas estas corrientes confluyen de manera inequívoca en lo que se conoce como marketing online, una noble disciplina, aún no reconocida, que justamente nuclea a diversas técnicas, tomadas de marketing y comunicación social, y extrapoladas a lo que es internet, con base en redes, pero también en blogs, foros, y hasta comunidades wikis como Wikipedia, por ejemplo. Como ya adelantamos, estas disciplinas pro-

pugnan la idea de la necesidad de planificar de manera estratégica un plan de comunicación, teniendo en cuenta a qué porción de mercado se busca llegar, qué tipo de producto o servicio se quiere difundir, y cómo la experiencia y opinión de los usuarios retroalimenta lo planificado, hasta incluso modificarlo.

Y seguramente en este punto se comenzarán a preguntar, ¿de qué sirve todo esto para un webmaster? Si tenemos en cuenta que un emprendimiento web, sea un sitio, un portal, o una página que sirva para ofrecer un servicio es un producto en sí, estamos en condiciones de afirmar que todos esos conceptos son plausibles de ser aplicados en la difusión de una web, con las modificaciones y adaptaciones necesarias de cada caso. En ello reside la importancia de estas teorías, y las redes sociales como herramientas de difusión, el hecho de que podemos utilizarlas para comunicar lo que hacemos, y asimismo que podemos generar una nueva idea del feedback que obtenemos al darlas a conocer. En la presente nota intentamos repasar de manera general tanto los principales conceptos de social media, o cuál red social es la más apropiada para cada emprendimiento, como así también qué herramientas podremos encontrar en cada caso. La primera parada es casi obligatoria, y se basa estrictamente en la popularidad: Facebook es, actualmente y de seguro lo seguirá siendo por mucho tiempo más, la red social más popular, con alrededor de 600 millones de usuarios y un valor estimado en 65.000 millones de dólares. Fue creada en 2004 originalmente por Mark Zuckerberg como una red que permitiría conectar a los estudiantes de la universidad de Harvard, pero



Twitter.png Es la red más influyente a nivel mundial, cuyo crecimiento se aceleró a partir de la utilización que le dan estrellas de cualquier parte del globo. Aún así, no es todavía un gran negocio.

pronto fue expandiendo su campo de acción a otras universidades similares, hasta que en septiembre de 2006 se permitió el registro de cualquier persona mayor de 13 años que poseyera una cuenta de e-mail. Uno de los aspectos más relevantes de Facebook, hoy algo relegado, es la posibilidad de conectarse a distintas redes, ya sean educativas, geográficas o laborales, siendo esta la originaria premisa de la página.

En cuanto al tema que nos trae, la herramienta más notoria que ofrece Facebook para un emprendimiento web es la posibilidad de crear una fan page, una página de la cual cada usuario se puede hacer fan y seguir las actualizaciones de la misma. El

grupo es de una dinámica similar, aunque actualmente también se encuentra bastante relegado. Estas fan pages son utilizadas por personalidades, empresas o cualquier tipo de emprendimiento que busque difundir contenidos, acciones y eventos, como así también generar un contacto más cercano con el público que los sigue. Otra de las herramientas que permitió el desarrollo de Facebook es la posibilidad de que programadores externos pudieran incluir sus propias aplicaciones dentro de la página. Es lo que popularmente conocemos como juegos o encuestas de todo tipo, las cuales también nos permiten difundir lo que hacemos de manera viral. La tercera y última herramienta la constituyen los avisos patrocinados del sitio, los cuales funcionan de manera similar a Adwords, de Google, con una fuerte segmentación demográfica. Básicamente, una empresa puede generar un anuncio de manera simple, con texto e imágenes, contratando el servicio de manera directa desde el link <http://facebook.com/advertising>, el cual se mide por la cantidad de clicks que hacen los usuarios en el aviso,

o por la cantidad de veces que éste aparece en pantalla. El fuerte de estos avisos es la segmentación de acuerdo a qué porción de mercado se apunte, y la posibilidad de conectarlos a la opción "Me Gusta", una de las más populares del sitio.

La siguiente red dentro del itinerario es Twitter, que está considerada la segunda en cuanto a popularidad en la actualidad. Está basada en el concepto de microblogging, y se cree que posee más de 200 millones de usuarios. Si bien tiene una alta valoración comercial, sus ingresos rondarían el medio millón de dólares anuales, poco si se considera que fue tasada en casi 8.000 millones de dólares. Fue creada en marzo de 2006 por Jack Dorsey, Evan Williams y Biz Stone, con la base de los reconocidos sms, trasladados al formato de un sitio web. Si bien no posee la popularidad de Facebook, su fuerte reside en la fortísima influencia que logró dentro de los ámbitos de negocios y la política, especialmente, los cuales lo adoptaron de manera veloz.

Twitter, a diferencia de Facebook, no contempla herramientas o aplicacio-



Es actualmente la red social más popular, y la de mayor proyección, en base a sus 600 millones de usuarios. Se encuentra en plena etapa de desarrollo de un e-mail propio.

nes para terceros, sino que su propia dinámica y estructura le permite ser utilizada según la conveniencia de cada usuario o empresa. Algunos políticos lo utilizan como difusión de actos de campaña o propuestas de todo tipo, otros como forma de comunicar sus propias ideas, diversos medios de comunicación linkean notas de sus portales web, e incluso compañías como Personal y Movistar buscan gestionar los reclamos de sus clientes mediante tweets. Es por ello que lo principal a la hora de encarar una estrategia de difusión mediante Twitter es conocer bien sus diversas funciones, y tener bien en claro a qué público se quiere llegar. Eso determinará de qué manera usaremos las características que la red nos ofrece. Son mensajes en 140 caracteres, básicamente, y el usuario decide cómo los utiliza. También está extendido el uso de publicidad, ya sea promo-

cionando la cuenta de una personalidad reconocida, como también sponsoreando determinados tweets.

La tercera red social en popularidad y relevancia en la actualidad es LinkedIn, la cual es altamente especializada, ya que está orientada exclusivamente al ámbito laboral. Justamente esa es su fortaleza, como así también su debilidad. Nació como un sitio orientado al sector de los negocios, y fue lanzado en mayo de 2003, aunque recién en el año 2008 comenzó un proceso de expansión que la sitúa actualmente en los 100 millones de usuarios, y un formato completamente focalizado en abarcar los diferentes aspectos que conciernen el campo laboral. Su creador fue Dan Nye, y en ella tienen lugar tanto usuarios que cuelgan su perfil profesional, como las empresas más reconocidas y millonarias del mundo globalizado.

Al ser una página con ese nivel de especialización, las posibilidades de ser utilizada dependen de si el mercado al que se apunta lo comparte con la red social. Igualmente, más allá de esta limitación, LinkedIn ofrece varias funcionalidades para quien se adentra en ella. Actualmente posee un programa de publicidad dentro de la red, al que se puede acceder mediante el link <http://linkedin.com/advertising> que parte de los 10 dólares por día, aunque lamentablemente se encuentra completamente en inglés. En la página principal se pueden ver diferentes tipos de publicidades estilo banner en Flash. Además, se puede crear el perfil de una empresa en particular, y generar grupos basados en intereses académicos o en el perfil laboral de los usuarios. También tiene lugar para aplicaciones de terceros, pero en este caso, sólo aquellas que el sitio patrocina, además de tener en desarrollo un sistema de avisos clasificados laborales, que aún se encuentra en estado beta.

Hasta acá hemos visto las tres redes sociales predominantes en la actualidad. Pero no son las únicas, ni quizás las más útiles. Un ejemplo claro es MySpace, que hoy se encuentra en franca decadencia, pero que le permitió, y aún permite, a miles de grupos, bandas y solistas de todo tipo, y cualquier origen, tener un lugar para difundir tanto su música como todas sus actividades, siendo esta su principal utilidad en la actualidad. Fue creada por Tom Anderson y Chris DeWolfe en el año 2003, y fue la primera red social en alcanzar la masividad, especialmen-



Las fan pages de Facebook son utilizadas por personalidades, empresas o cualquier tipo de emprendimiento que busque difundir contenidos, acciones y eventos, como así también generar un contacto más cercano con el público que los sigue.”

te en el mercado norteamericano. Originariamente fue concebida para generar el intercambio de imágenes, música y videos entre usuarios, a los cuales se les asignaba una página de diseño básico, elegibles a través de distintas plantillas. Por el rumbo que tomó en los últimos años, sus mayores funcionalidades son mayormente utilizadas por la industria de la música, que la adoptó como propia. Más que nada, esto se debió a que permitía la difusión de música de manera gratuita, bajo un reproductor propio, aunque al mismo tiempo impedía cualquier descarga ilegal. Sus problemas comenzaron con el auge de Facebook, la cual la reemplazó paulatinamente en la consideración del público en general. Más allá de esto, aún tiene numerosas funcionalidades de gran utilidad, como por ejemplo los boletines periódicos, la posibilidad de abrir espacios para grupos, la creación de aplicaciones al estilo Facebook, el canal MySpace TV y los avisos MySpace Classifieds. La

contra principal que posee es su escasa penetración a nivel nacional, a pesar de los diferentes eventos que generaron con varios grupos de la escena musical argentina.

Existen muchas redes sociales más, aunque de mucha menor relevancia. Todas apuntan de manera determinada a distintos estratos de público, o de mercado, como cada uno prefiera. Entre las que cabe destacar, mencionamos a Sónico y Tuenti, similares a Facebook pero para el público latinoamericano y español, respectivamente; Tumblr, que se maneja bajo un formato similar a Twitter, pero permite la publicación de imágenes, audios, links y citas de todo tipo; Flickr, para publicar todo tipo de imágenes, y con gran difusión dentro del rubro de la fotografía; y Formspring, basada en el formato de preguntas y respuestas, que puede ser articulado con otras redes más populares. Como se ve, existen muchas y para todos los gustos, sumadas a las que no fueron nombradas, y todas poseen algún tipo de herramienta para alcanzar la meta deseada, que es llegar de la mejor manera al público. Es cuestión de identificar a qué grupos sociales se pretende alcanzar, y cuál es la mejor red social para lograr ese objetivo. **P**

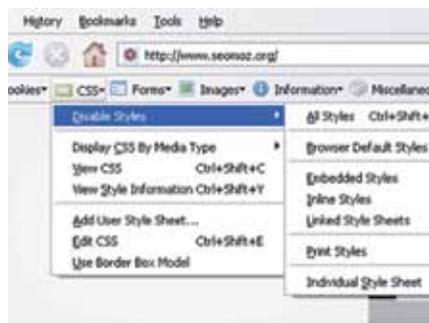


Extensiones para Firefox indispensables para desarrolladores web

Luis Altamiranda
luis.altamiranda@pixelscode.com

Las extensiones de Firefox son pequeños agregados al navegador que le dan mayor funcionalidad. Existen de todo tipo y para hacer prácticamente cualquier cosa. En esta oportunidad nos centraremos en aquellas que pueden resultarnos útiles a la hora de desarrollar nuestro sitio web. De esta manera, encontraremos extensiones para probar nuestra página como así también para “espiar” las páginas de los demás y ver (y de ese modo tratar de comprender) cómo están diseñadas para así sacar ideas.

WEB DEVELOPER



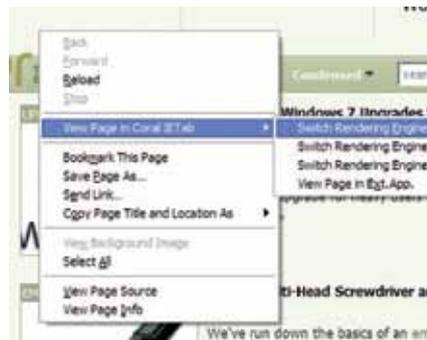
Autor: chrispederick
URL: <https://addons.mozilla.org/es-ES/firefox/addon/60>

Sin dudas es “la extensión” imprescindible para todo desarrollador web. Se presenta al usuario como una barra de botones en donde cada uno abre un menú con múltiples opciones para elegir. Así, por ejemplo, en el botón CSS, la opción Ver información de los estilos nos permitirá ver detalladamente la configuración de cada elemento CSS. O, en el botón Resaltar, podremos destacar los distintos elementos que componen la página que estamos visitando y así poder ver cómo está diagramada.

Otra de las extensiones estrella para los desarrolladores. Con una interfase limpia y clara, al presionar la tecla F12 nos abrirá una ventana en la parte inferior del navegador. Entre sus opciones podemos destacar en el apartado Red la posibilidad de medir las peticiones que hace una determinada página para cargarse, mostrando el tiempo de carga de cada elemento al momento en que comienza cada petición.

como un pequeño ícono en la derecha de la barra de estado que cambia del de Firefox al del Internet Explorer de acuerdo al motor de navegador que esté siendo utilizado. Si hacemos clic sobre el mismo, carga nuevamente la pestaña actual con el otro motor, pero si hacemos clic con la rueda del mouse (lo que sería el botón central), abre una nueva pestaña con la página vista por el otro motor de navegación.

CORAL IE TAB



FIREFTP



FIREBUG



Autor: Joe Hewitt, johnjbarton, robcee, FirebugWorkingGroup
URL: <https://addons.mozilla.org/es-ES/firefox/addon/1843>

Autor: quaful
URL: <https://addons.mozilla.org/es-ES/firefox/addon/10909>

Esta extensión nos permite con un simple clic cargar dentro del mismo Firefox la página actual, pero vista con el motor del navegador Internet Explorer. De este modo, podremos verificar la manera en que se visualiza una misma página en ambos navegadores. Aparece

Autor: Mime Cuvalo
URL: <https://addons.mozilla.org/es-ES/firefox/addon/684>

Excelente extensión que nos trae un completo cliente de FTP directamente a la ventana de nuestro navegador. Con la tradicional interfase de dos paneles típica de los clientes FTP, podremos conectarnos a cualquier servidor FTP y ya sea descargar o subir archivos al mismo. También soporta "arrastrar



y soltar” entre ambos paneles, lo que lo hace sumamente intuitivo al momento de utilizarlo.

LINKCHECKER

Relative Links

- Link to a **sub-folder**
- We all have our **oops**
- Backing out to **another folder**
- Stay in the **current folder**
- **Two slashes** are better than one (thanks Slashdot)

Absolute Links

- An **absolute link**
- Gotta try a **dead absolute link**
- Don't forget **server errors**
- **FTP addresses** are addresses too

Other Links

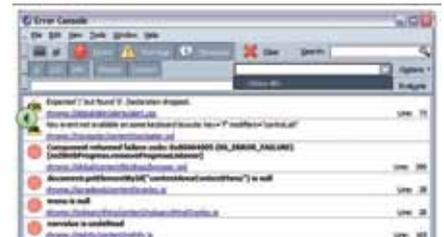
- Link to a **non-existent domain name**
- A **dead link** and a **valid link** to a **different site**

Autor: Kevin Freitas

URL: <https://addons.mozilla.org/es-ES/firefox/addon/532>

Esta simple pero muy útil extensión nos permitirá verificar si los links del sitio que estamos visitando son válidos o no. Ideal para quienes son encargados de sitios web con links a sitios que no nos pertenecen. Con un simple clic en el menú Herramientas, la opción Check Page Links irá recorriendo los enlaces de la página y “pintándolos” de verde o rojo según sean o no válidos.

CONSOLE²

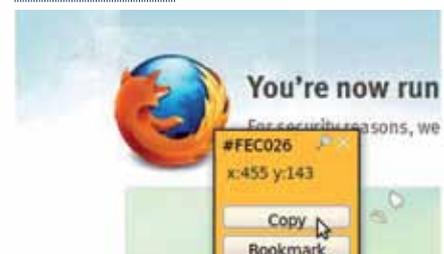


Autor: Philip Chee, zeniko

URL: <https://addons.mozilla.org/es-ES/firefox/addon/1815>

Otra extensión dedicada exclusivamente para los desarrolladores web, ofrece mejoras en la consola de errores de nuestro navegador. Utilizándola podremos, por ejemplo, filtrar los errores según se traten de CSS, JavaScript, XML, etc.

RAINBOW



Autor: harthur

URL: <https://addons.mozilla.org/es-ES/firefox/addon/14328>

Con un ícono representando un arco iris en la barra de estado del navegador, esta extensión nos permitirá consultar los colores utilizados en la página que estemos visitando. Entre sus opciones, además de poder elegir un elemento con el mouse y que nos informe el código del color (en el formato que deseemos; RGB, hexadecimal, HSL) puede analizar la página y mostrarnos los colores según las especificaciones CSS que esté utilizando.

“ Encontraremos extensiones para probar nuestra página como así también para “espíar” las páginas de los demás y ver (y de ese modo tratar de comprender) cómo están diseñadas para así sacar ideas.”

MEASUREIT



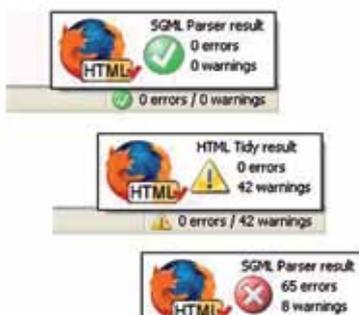
Autor: Kevin Freitas

URL: <https://addons.mozilla.org/es-ES/firefox/addon/539>

Parece algo trivial, pero muchas veces a la hora de diseñar un sitio nos interesa saber cuánto mide tal o cual objeto en la pantalla. Para eso, esta extensión nos será extremadamente útil. Basta con activarla desde el botón correspondiente para que nuestro puntero se convierta en una práctica regla con la que podemos medir en alto y ancho

en píxeles lo que sea que veamos en la página cargada en el navegador.

HTML VALIDATOR



Autor: Marc Gueury

URL: <https://addons.mozilla.org/es-ES/firefox/addon/249>

Cumplir con los estándares HTML es algo fundamental en el desarrollo web de quienes realicen un proyecto prolijo y entendible. Esta extensión, que se aloja en un ícono en la barra de estado, nos indicará si el sitio que estamos visitando cumple con

estas reglas o no. Más allá de eso, si hacemos doble clic en el mismo nos abrirá un completo informe mostrándonos el código fuente de la página y los errores y advertencias que ha detectado. Incluye una completa ayuda para que podamos comprender estos mensajes y corregirlos.

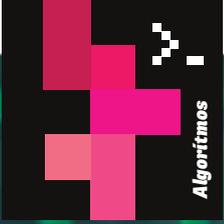
CODEBURNER FOR FIREFOX



Autor: SitePoint

URL: <https://addons.mozilla.org/es-ES/firefox/addon/13048>

Con Firefox también podemos aprender... Y para ello es esta extensión, una poderosa herramienta de referencia de código HTML y CSS. Aparece en la barra de estado en la parte derecha como un campo de búsqueda o si no haciendo clic en el ícono correspondiente. Podremos buscar cualquier término que nos mostrará las coincidencias, un rápido resumen y la compatibilidad del mismo en los distintos navegadores. Luego, si así lo deseamos, podemos acceder a una información mucho más detallada y con ejemplos de código para que podamos comprender mejor su utilización. **P**



Algoritmos: Insertion sort

Por

@pixelscode.com
@



Continuando con la serie iniciada en la edición anterior, en esta oportunidad vamos a analizar el algoritmo de ordenamiento conocido como “Insertion Sort”. Comparándolo con el Ordenamiento de Burbuja que ya hemos visto, notaremos que comparte todas sus ventajas, pero no posee la mayoría de sus limitaciones, por lo cual dominarlo a la perfección es una herramienta muy útil para los programadores de cualquier lenguaje. Además, muchos lo consideran el método de ordenamiento más natural, ya que es muy similar al sistema empleado por los seres humanos al aplicarlo sobre situaciones cotidianas, como por ejemplo ordenar un mazo de naipes.

SORT POR COMPARACIÓN

Al igual que el Bubble Sort, el Insertion Sort es un algoritmo de ordenamiento “por comparación”, es decir, trabaja comparando cada valor a ordenar con los demás valores de la lista. Sin embargo, el método de burbuja pertenece a la categoría de algoritmos de sorting “de intercambio”: toma pares de elementos de una lista y de ser necesario, intercambia la posición de los integrantes de ese par. El ordenamiento por inserción, en cambio, va recorriendo la lista de valores a ordenar, tomando de a un elemento e insertándolo en la posición correcta.

FUNCIONAMIENTO DEL ALGORITMO

En cada iteración, se toma arbitrariamente un elemento de la lista a ordenar. En la primera pasada, con el elemento seleccionado se forma una nueva lista, que al contar con un único elemento ya está ordenada. En las pasadas posteriores, se va tomando con el mismo criterio un nuevo elemento de la lista a ordenar, e insertándolo en la lista ya ordenada de la siguiente forma: se reco-

re la lista ordenada hasta encontrar un valor mayor, insertando el elemento nuevo en la posición anterior a la del valor hallado. En la implementación aquí propuesta no hace falta crear una segunda lista explícitamente, sino que se “subdivide” la lista sobre la que se trabaja, quedando tras cada iteración en los primeros lugares los elementos ya ordenados, y en los últimos los que restan ordenar. Para que se entienda mejor, vamos a aplicar el método a un ejemplo concreto. Partiendo de una lista cuyos elementos son [2, 1, 4, 3]:

- Al comenzar, tenemos una lista ordenada conformada por el primer elemento de la lista original: [2], y restan en esta tres elementos a ordenar: [1,4,3].

- Tomamos el primer elemento de la lista a ordenar: 1. El primer (y por el momento único) elemento de la lista ya ordenada es 2, por lo que ubicamos el elemento 1 antes de éste. Ahora tenemos en la ya ordenada [1,2] y en la que resta ordenar [4,3].

- Continuamos con el siguiente elemento de la segunda lista: 4. Al no encontrar ningún valor mayor que este en la lista ya ordenada, queda ubicado al final, obteniendo [1, 2, 4] y restando en la lista por ordenar un único item, [3].

- Finalmente el elemento restante de la lista a ordenar, 3, es insertado en la lista nueva en la posición anterior al primer valor encontrado que sea mayor que éste, obteniendo como resultado final la lista ordenada: [1, 2, 3, 4].

PSEUDOCÓDIGO

En pseudocódigo, el algoritmo de ordenamiento por inserción según lo expuesto puede escribirse así, asumiendo que trabajamos con índices de base cero (es decir, el número de índice del primer elemento es 0, el segundo es 1 y así sucesivamente):

Dada una lista L, de N elementos

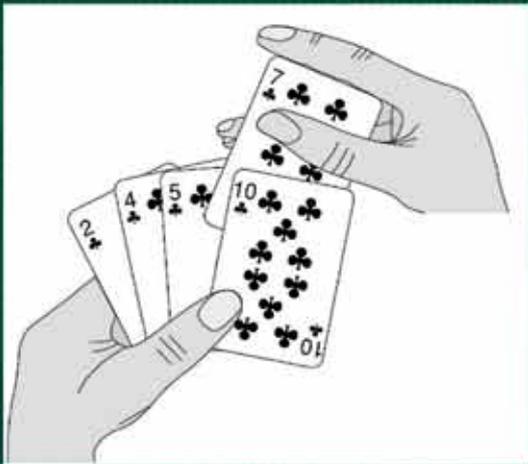
```

for i = 0 a N - 1
    elemento = L(i)
    x = i - 1
    mientras x >= 0 y L(x) > elemento, repetir:
        L(x + 1) = L(x)
        x = x - 1
    fin mientras
    L(x + 1) = elemento
fin for

```

IMPLEMENTACIÓN EN PYTHON

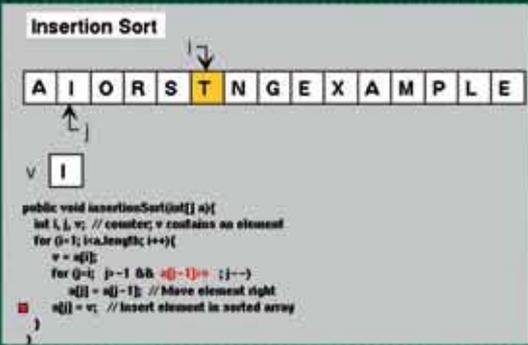
Nuevamente recurrimos a Python para realizar como ejemplo una implementación funcional de nuestro algoritmo, ya que por su clara sintaxis, el programa Python es muy similar al pseudocódigo. Además, como ya remarcamos en otras ocasiones, se trata de un lenguaje de programación que es muy fácil de aprender y utilizar.



.....

Cuando un ser humano ordena las cartas de una baraja en orden ascendente o descendente, aplica exactamente el mismo concepto en el que se basa el algoritmo de Insertion sort.

.....



.....

Una implementación del ordenamiento por Inserción escrita en lenguaje Java.

.....

```
# -*- coding: cp1252 -*-
L = [6, 4, 5, 2, 3, 1]
N = len(L)
itera = 0

print "Lista =", L, "elementos =", N

for i in range(N):
    valor = L[i]
    x = i - 1
    desplaza = 0

    while x >= 0 and L[x] > valor:
        desplaza = desplaza + 1
        L[x + 1] = L[x]
        x = x - 1

    L[x + 1] = valor
    print "Ordenando", valor, "Desplazado", desplaza, "posiciones"

    itera = itera + 1
    print "iteración =", itera, "i =", i, "-->", L
```

El resultado de la ejecución de este programita Python presenta la siguiente salida por pantalla:

```
Lista = [6, 4, 5, 2, 3, 1] elementos = 6
Ordenando 6 Desplazado 0 posiciones
iteración = 1 i = 0 --> [6, 4, 5, 2, 3, 1]
Ordenando 4 Desplazado 1 posiciones
iteración = 2 i = 1 --> [4, 6, 5, 2, 3, 1]
Ordenando 5 Desplazado 1 posiciones
iteración = 3 i = 2 --> [4, 5, 6, 2, 3, 1]
Ordenando 2 Desplazado 3 posiciones
iteración = 4 i = 3 --> [2, 4, 5, 6, 3, 1]
Ordenando 3 Desplazado 3 posiciones
iteración = 5 i = 4 --> [2, 3, 4, 5, 6, 1]
Ordenando 1 Desplazado 5 posiciones
iteración = 6 i = 5 --> [1, 2, 3, 4, 5, 6]
```

Cada posición desplazada representa una iteración del bucle interno "while". Para obtener la cantidad total de iteraciones realizadas por el algoritmo deben sumarse las iteraciones del bucle externo "for" a la cantidad total de posiciones desplazadas.

ASPECTOS TEÓRICOS

Como puede apreciarse al analizar el pseudocódigo, el ordenamiento por inserción tiene diversas ventajas. Es sencillo de comprender e implementar, si bien es algo más complejo que el de burbuja. Pero al igual que en este último, su rendimiento es malo para listas muy largas. Esto se debe a que su complejidad (cantidad de "pasadas" necesarias para completar el orde-

namiento) en el peor de los casos -que para este algoritmo sería una lista con el orden inverso al deseado-, es de $O(N^2)$, la misma que para el Bubble Sort. Y también como el “burbujeo” es altamente efectivo para listas que no posean grandes cantidades de elementos, pero lo supera claramente en el caso de listas con un “desordenamiento” intermedio, o que ya se encuentren ordenadas, dado que allí su complejidad desciende a $O(N)$. La complejidad exacta de una lista determinada es de $O(N + d)$, donde d es igual al número total de desplazamientos realizados. De hecho, el Insertion Sort es considerado el algoritmo ideal para listas de pocos elementos, a punto tal que otros algoritmos más complejos, de rendimiento superior para listas largas, como el famoso “Quick Sort”, incorporan como optimización la aplicación del ordenamiento por inserción cuando la lista posee una baja cantidad total de elementos. Una pregunta válida para hacerse es a partir de cuántos elementos la aplicación de este algoritmo deja de ser eficiente en cuanto a tiempos de respuesta. Aunque la medida depende del poder de procesamiento del equipo que lo vaya a ejecutar, es habitual su empleo en listas de hasta 10 ítems.

OPTIMIZACIONES POSIBLES

Entre las distintas optimizaciones propuestas para este algoritmo, la más difundida es conocida como “Shell Sort”, por ser su creador el norteamericano Donald L. Shell. Su versión del ordenamiento fue concebida en 1959, como parte de los trabajos realizados para obtener su Ph. D. en matemáticas en la Universidad de Cincinnati. La principal ventaja de esta versión es que reduce la cantidad total de desplazamientos. La idea de Shell fue modificar el algoritmo para que en lugar de analizar elementos contiguos, se realice la comparación entre elementos separados por varias posiciones, reduciendo la distancia entre los valores a comparar en cada pasada. De esta manera, los elementos se desplazan más rápidamente hacia su lugar en la lista ordenada. Según la implementación original de D. Shell, la cantidad de espacios entre elementos en la primera iteración debe ser la equivalente a la cantidad de elementos de la lista dividido dos, redondeada al valor entero menor más cercano al resultado de la división. Tras cada iteración del bucle principal, se reduce el gap (distancia) entre elementos a comparar, dividiéndolo nuevamente por 2 y redondeándolo hacia abajo al entero más cercano, hasta llegar a la distancia de 1. En esta instancia trabajará como el Insertion sort tradicional. Veámoslo en pseudocódigo:

Dada una lista L, de N elementos

```
dist = int(N / 2)
mientras dist > 0, repetir:
  for i = dist a N - 1
    elemento = L(i)
    x = i
    mientras x >= dist y L(x - dist) > elemento, repetir:
      L(x) = L(x - dist)
      x = x - dist
    fin mientras
    L(x) = elemento
  fin for
```



```

[9, 7, 5, 6, 4, 2, 1, 3, 8]
[7, 9, 5, 6, 4, 2, 1, 3, 8]
[5, 7, 9, 6, 4, 2, 1, 3, 8]
[5, 6, 7, 9, 4, 2, 1, 3, 8]
[4, 5, 6, 7, 9, 2, 1, 3, 8]
[2, 4, 5, 6, 7, 9, 1, 3, 8]
[1, 2, 4, 5, 6, 7, 9, 3, 8]
[1, 2, 3, 4, 5, 6, 7, 9, 8]
[1, 2, 3, 4, 5, 6, 7, 8, 9]

```

En esta imagen vemos claramente el funcionamiento de la implementación estándar de Insertion Sort. El elemento coloreado es el primer ítem de la lista desordenada, mientras que a su izquierda se va conformando tras cada pasada la lista ordenada. Para lograrlo hacen falta dos bucles: uno externo, que haga tantas iteraciones como elementos tenga la lista a ordenar; y uno interno, que para cada pasada del bucle externo realice las iteraciones que se requieran para desplazar el elemento tratado a su posición correcta en la lista ordenada.

```

dist = int(dist / 2)
fin mientras

```

La implementación Python del Shell Sort sería así:

```

# -*- coding: cp1252 -*-
L = [6, 4, 5, 2, 3, 1]
N = len(L)
dist = int(N / 2)
itera = 0

print "Lista =", L, "elementos =", N

while dist > 0:
    for i in range(dist, N):
        valor = L[i]
        x = i
        desplaza = 0

        while x >= dist and L[x - dist] > valor:
            desplaza = desplaza + dist
            L[x] = L[x - dist]
            x = x - dist

        L[x] = valor
        print "Ordenando", valor, "Desplazado", desplaza, "posiciones"

    itera = itera + 1
    print "iteración =", itera, "i =", i, "-->", L

    dist = int(dist / 2)

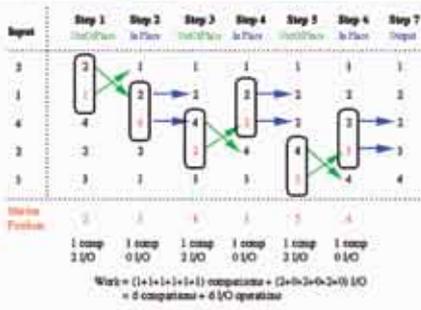
```

Y la salida por pantalla:

```

Lista = [6, 4, 5, 2, 3, 1] elementos = 6
Ordenando 2 Desplazado 3 posiciones
iteración = 1 i = 3 --> [2, 4, 5, 6, 3, 1]
Ordenando 3 Desplazado 3 posiciones
iteración = 2 i = 4 --> [2, 3, 5, 6, 4, 1]
Ordenando 1 Desplazado 3 posiciones
iteración = 3 i = 5 --> [2, 3, 1, 6, 4, 5]
Ordenando 3 Desplazado 0 posiciones
iteración = 4 i = 1 --> [2, 3, 1, 6, 4, 5]
Ordenando 1 Desplazado 2 posiciones
iteración = 5 i = 2 --> [1, 2, 3, 6, 4, 5]
Ordenando 6 Desplazado 0 posiciones
iteración = 6 i = 3 --> [1, 2, 3, 6, 4, 5]
Ordenando 4 Desplazado 1 posiciones
iteración = 7 i = 4 --> [1, 2, 3, 4, 6, 5]
Ordenando 5 Desplazado 1 posiciones
iteración = 8 i = 5 --> [1, 2, 3, 4, 5, 6]

```



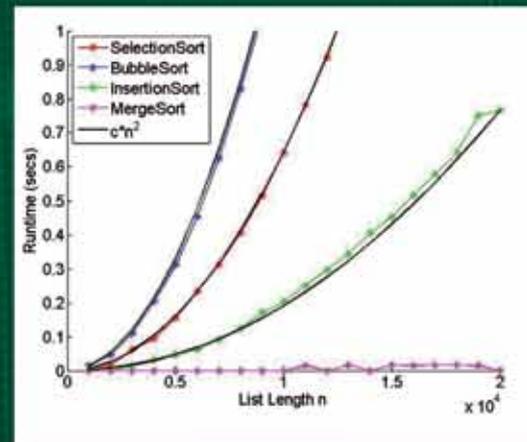
Aquí observamos como con la versión clásica del Insertion Sort los elementos se desplazan en el bucle interno de a una posición a la vez.

Aunque la cantidad total de iteraciones del bucle externo FOR es mayor, se reducen sensiblemente las del bucle interno WHILE. En este ejemplo, en la primera iteración se desplazó el elemento “3” tres posiciones hacia el comienzo de la lista en una sola pasada del bucle interno, por lo cual la cantidad total de pasadas del bucle WHILE es mucho menor, resultando en el peor de los casos posibles una complejidad de $O(N \log^2 N)$.

Otra optimización muy innovadora, de desarrollo reciente, es la conocida como “Library Sort”, publicada por Michael A. Bender, Martín Farach-Colton y Miguel Mosteiro en 2006. La idea es básicamente dejar “espacios vacíos” entre los elementos de la lista ordenada, para reducir la cantidad total de desplazamientos en listas con mayor cantidad de elementos, de la misma manera que lo haría, como ejemplifican los autores, un bibliotecario ordenando libros por orden alfabético. Si completara un estante en su totalidad, y luego tuviera que agregar un libro más, se vería obligado a desplazar a otro estante todos los libros cuyo título sea subsiguiente en el orden al nuevo. Si se dejan algunos espacios vacíos entre los libros al realizar los primeros ordenamientos, al agregar un nuevo elemento se evita una gran cantidad de desplazamientos para hacer lugar a este nuevo ítem. Se trata de un concepto complejo, que escapa a los alcances de esta nota, pero quienes estén interesados pueden leer el “paper” de Bender, Farach-Colton y Mosteiro, disponible para descargar en formato PDF en la dirección [<http://www.cs.sunysb.edu/~bender/newpub/BenderFaMoo6-librariansort.pdf>].

EN CONCLUSIÓN

Como hemos visto, las ventajas del ordenamiento por inserción sobre el ordenamiento de burbuja presentado en la edición anterior son considerables, pero también basándose en conceptos sencillos, muy fáciles de comprender e implementar en cualquier lenguaje. Sin embargo, aunque es considerablemente más eficiente que otros algoritmos de ordenamiento cuadráticos, sigue siendo una opción pobre para listas compuestas por una gran cantidad de elementos. En las próximas ediciones, seguiremos explorando otros tipos de algoritmos de ordenamiento, agregando nuevos “usos” a la “navaja suiza” que constituye para el buen desarrollador de software, sea cual sea el lenguaje de programación de su elección, el conocimiento sobre el funcionamiento de los algoritmos “clásicos”. Con este término nos referimos a aquellos que se enseñan en el ámbito académico desde hace muchas décadas. Recordemos que además de ser útiles en sí mismos cuando tengamos que trabajar ordenando listas, arreglos o vectores, también contribuyen a formar el pensamiento abstracto, indispensable para encarar analíticamente la solución de un problema y poder expresarla a través de la sintaxis de cualquier lenguaje de programación. **P**



En el gráfico se demuestra la mayor eficacia en tiempos de ejecución de Insertion Sort (curva verde) respecto del Bubble Sort (curva azul), sobre todo a medida que crece el tamaño de la lista a ordenar. Ambos son superados ampliamente por Merge Sort, que veremos más adelante, cuyos tiempos de corrida apenas se ven afectados por el aumento exponencial del tamaño de la lista.



Tu website móvil

Pensar en grande para dispositivos chicos





Que Internet llegó a los teléfonos móviles no es una novedad pero que son cada vez más las compañías que quieren ver bien su página en ellos, sí lo es. Veremos cómo participar del mercado de portales hechos a la medida de las necesidades.

Gentos de personas han incluido entre sus pertenencias imprescindibles al pequeño, ultra liviano y multimedial celular. Esto es porque, el aparato que es cada vez más práctico, ofrece una gran variedad de funciones que lo hacen más completo que un escritorio de oficina o, según el equipo, una biblioteca. En este sentido, los contenidos se han adecuado hasta convertir un simple teléfono en un dispositivo anywhere-anytime.

Por lógica, si cada vez más usuarios tienen uno de estos, con el tiempo son más los contenidos que se ofrecen por esta vía. Empresas, entidades de servicio, instituciones públicas o privadas buscan entrar en una pantalla de unas pocas pulgadas con un menú que indica a una persona común qué necesita y cómo adquirirlo.

Por estos motivos, los desarrolladores y diseñadores tienen que estar atentos a este flujo de sitios web que no se verán en pantallas LCD o monitores de alta gama, sino en equipos con menor resolución, baja velocidad, poca memoria y, en algunos casos, un reducido espacio para esbozar un slogan o un diseño.

Desde hace casi una década existe un protocolo de aplicaciones inalámbricas que se encarga de que el acceso a páginas web desde los dispositivos móviles sea cada vez más viable. WAP, Wireless Application Protocol, es la tecnología que facilita que los celulares sean indispensables en la vida de los usuarios, y que también es aplicable a otros dispositivos inalámbricos con conexión a Internet, como las netbook o Palm.

Las compañías Nokia, Phone.com, Motorola y Ericsson fueron las responsables de que se enciendan los primeros portales WAP al crear el WAP Forum con el cual se empezó a definir una sintaxis, variables y elementos en wml y para lo cual otros fabricantes se sumaron desarrollando capacidades adicionales. Su sitio oficial es www.wapforum.com

EN POCAS PALABRAS

WAP es un protocolo abierto que permite interactuar con otros servicios de diferentes vendedores, siempre y cuando se puedan aplicar a este funcionamiento. Está creado específicamente para utilizarlo en equipos portátiles con conexión

a Internet y ofrece la facilidad de acceder a esos contenidos por medio de micro-browser de manera más rápida. Que esto sea así dependerá tanto del desarrollo como del diseño.

Si bien no requiere de un gran esfuerzo de programación y es tanto o más fácil que armar una página web tradicional en HTML, la contrapartida es que a la hora del armado hay que concentrarse en no olvidar una coma, ni un espacio puesto que podríamos perder un buen tiempo tratando de resolver una página de error. Por lo demás, los portales WAP son, podría decirse, una versión reducida de los website convencionales.

FUNCIONAMIENTO WAP

La transmisión de información con este protocolo pretende ser más sencilla, fácil y eficaz para equipos inalámbricos. En primera instancia al recurrir a un comando WAP se realiza un pedido de URL, el dispositivo envía esta URL a un proxy WAP donde lo convierte en un http que redirecciona a un servidor Web. Este devuelve una página WML, en donde el proxy WAP



Si bien no requiere de un gran esfuerzo de programación y es tanto o más fácil que armar una página web tradicional en HTML, la contrapartida es que a la hora del armado hay que concentrarse en no olvidar una coma, ni un espacio puesto que podríamos perder un buen tiempo tratando de resolver una página de error."

“Actualmente las imágenes se pueden visualizar en formato WBMP que se descargan más rápido. Para generarlas es necesario valerse de un plug-in de Adobe PhotoShop, o un convertidor de BMP a WBMP que, por caso, el software de Nokia trae consigo.”

se encarga de recibirla y convertirlo a formato binario para que finalmente llegue al móvil, en donde se procesa y exhibe al usuario.

Para su creación se utilizará un metalenguaje llamado WML (Wireless Markup Language) que no es más que un derivado del HTML (Hyper-Text Markup Language), pero que se servirá del dialecto WMLScript en lugar de Javascript, y que podremos ver que será para el caso de los portales WAP la nueva denominación de las extensiones web (.wml).

Cualquier editor de texto es lo suficientemente apropiado para generar una página principal a la cual se unirán cuantos enlaces sean los necesarios. Recordemos que la pantalla no nos permitirá extendernos demasiado y que a esto se le suma que normalmente los usuarios no navegan mucho tiempo en sus celulares, por esta razón deben ser indicaciones simples y directas. Puestos a construir el portal y a modo

de receta conviene tener en cuenta las partes que integran el proceso de construcción de un portal WAP, estas son: una plataforma, un servidor web y una base de datos, una interfaz administrativa, un simulador para las pruebas locales y otro para las finales.

Como realizar todas las pruebas en un celular serían muy costosas, se puede recurrir a una diversidad de emuladores antes de trasladarlo al móvil. Así es como webtaller.com recomienda para Windows el WinWAP y para Linux el Waplet. De todas maneras, es preciso realizar varias pruebas en emuladores para asegurar los resultados en diferentes modelos de móviles.

HACIA UN PORTAL 2.0

La tecnología WAP está destinada a responder a las necesidades del cliente móvil, éste solicita un contenido al servidor y el mismo se lo proporciona. Estos beneficios están maximizados por el aporte de la telefonía que sirve como el medio para que el usuario lleve toda la información que precisa en su bolsillo.

La creciente utilización de teléfonos móviles, la participación cada vez más activa de las personas en las redes, el consumo motivado por medio de Internet y este como una fuente de información privilegiada, son razones suficientes para que el desarrollo de portales WAP se perfeccione con el tiempo.

SITUACIÓN DEL MERCADO MÓVIL

- Existen más de 2.450.000.000 de conexiones móviles mundiales.
- La descarga de contenidos para móviles (imágenes, ringtones, videos, logos) se ha extendido entre más de un 70 % sólo entre los españoles.
- El mercado español de contenidos digitales para teléfonos móviles pasó

de 875 millones de euros en 2005 a cerca de 3200 millones en 2010.

- Un 35% de internautas españoles acuden a portales WAP de telefonía para personalizar sus móviles, esto representa cerca de los 4,5 millones de usuarios.

(FUENTE 3GSM)

El notable crecimiento del mercado obligó al grupo WAP Forum a satisfacer las necesidades de los nuevos usuarios. Unos años después se creó la versión 2.0 que agregó soporte para los distintos protocolos de transmisión de datos garantizando con ello mayor conectividad y amplió la cantidad de aplicaciones juntamente con la llegada de nuevas operadoras y usuarios. Pese a que las primeras versiones ofrecían la posibilidad de funcionamiento de imágenes, pasó bastante tiempo hasta que se pudo reducir la cantidad de bytes que consume en la descarga.

Actualmente las imágenes se pueden visualizar en formato WBMP que se descargan más rápido. Para generarlas es necesario valerse de un plug-in de Adobe PhotoShop, o un convertidor de BMP a WBMP que, por caso, el software de Nokia trae consigo.

La llegada de Wap 2.0 brinda otras características importantes como el proceso de comunicación por http/1.1, donde no media un proxy. Aunque es recomendable trabajar con proxy WAP ya que genera un empuje más importante con una notable velocidad de navegación.

LOS NUEVOS SERVICIOS 2.0

Estas son algunas de las aplicaciones nuevas que aporta la flamante versión:

WAP Push: Este servicio permite enviar el contenido a dispositivos,



utilizando aplicaciones basadas en servidores utilizando el proxy. Esto permite enviar mensajes a usuarios, alertas de tráfico, etc.

User Agent Profile: Capta las capacidades del cliente y las preferencias de los usuarios.

Wireless Telephony Applications: Brinda las herramientas para el manejo de aplicaciones avanzadas para ser operadas con aplicaciones tradicionales que soporten el manejo de datos.

External Functionality Interface: Especifica la interfase entre WAE y componentes o entidades que se ejecutan fuera de las capacidades WAE. Lo cual sería análogo a los Plug-In que se utilizan con los browser normales.

Persistent Storage Interface: Define un estándar para servicios de almacenamiento, que está asociado con una interfase bien definida para or-

ganizar, acceder, almacenar y acceder datos en un dispositivo inalámbrico u otros dispositivos que conectados a la memoria.

Data Synchronization: Es una solución para la transmisión de datos de manera sincronía.

Multimedia Messaging Service: Provee un servicio para efectuar una futura implementación de mensajes.

Pictogram: Permite utilizar pequeñas imágenes.

DE TU WEB AL WAP

Ante un mercado en pleno crecimiento de portales Wap en donde muchos usuarios no desean pasar por las etapas de desarrollo y producción, pero sí quieren ver sus sitios en las pantallas de pocas pulgadas, existen quienes se encargan de llevar la web a los dispositivos móviles en sencillos pasos.

Cada vez más compañías que ofrecen servidores proporcionan ahora todas las herramientas para que los sitios sean efectivos y lleguen a sus clientes de forma más rápida y con todas las prestaciones que estos requieren. Una de las pioneras en implementar los portales WAP en el mercado argentino es Dattatec.com con su servicio DattaMóvil.

El mismo permite a sus usuarios obtener un sitio adaptado a dispositivos móviles con mínimo esfuerzo y sin necesidad de tener conocimientos de diseño o programación. De esta forma los websites que no están preparados para los móviles pueden aprovechar el tráfico de los visitantes que navegan por estos dispositivos.

¿CÓMO FUNCIONA?

DattaMóvil demandó siete meses de producción de un equipo integrado por personas especializadas en



En total un grupo de ingenieros, diseñadores, front-end developers, programadores y analistas de usabilidad de Dattatec.com enfrentaron el propósito de hacer realidad los portales WAP para el mercado argentino.”

Marketing y Desarrollo Estratégico e Investigación y Desarrollo para cuestiones que tienen que ver con infraestructura tecnológica y compatibilidad con otros productos de la compañía. En total un grupo de ingenieros, diseñadores, front-end developers, programadores y analistas de usabilidad enfrentaron el propósito de hacer realidad los portales WAP para el mercado argentino.

Al contratar el servicio se obtiene una versión del sitio preparada para móviles y un acceso a un Back-End disponible para modificar el contenido y algunos aspectos visuales las veces que lo crea necesario. Cuando el dominio de un usuario de DattaMóvil recibe una visita, éste determina en forma automática si debe mostrar el sitio habitual (no móvil) o el sitio preparado para dispositivos móviles en función del equipo que el visitante utiliza para navegar.

Lo interesante de la propuesta de DattaMóvil es que, a comparación de otros servicios, los visitantes pueden acceder a su sitio desde su dominio sin recurrir a direcciones especiales. Actualmente, muchas empresas dependen del

funcionamiento optimizado de sus websites para llegar a sus clientes y si estos acceden cada vez con más facilidad a Internet y usan dispositivos móviles, todo indica que el gran negocio actualmente cabe en la cartera de la dama o en el bolsillo de un caballero. ¿Te vas a quedar afuera?

DATOS QUE SUMAN

Internet genera cada vez más oportunidades de negocios dentro de la nube. Aquí veremos un informe de usuarios de dispositivos móviles con capacidad para conectarse a Internet:

- * 5 billones de usuarios móviles en 2010.
- * 1 billón de conexiones a redes móviles capaces de conectar los dispositivos a internet en 2010.
- * Para 2013, los usuarios que visitarán los sitios a través de móviles excederán los 1.8 billones.
- * Para 2013, las visitas a sitios web desde móviles superarán a las provenientes de PCs (Morgan Stanley)
- * Hacia 2012, las búsquedas a través de móviles crecerán del 9% al 20% (Nielsen)
- * Abril de 2010 mostró 33.5 búsquedas

en mapas a través de móviles, un salto del 44%

(Fuente: comScore)

CLAVES DEL ÉXITO EN EL DISEÑO WAP

- Funcionalidad: El usuario móvil busca una información precisa y esta debe ser entregada sin rodeos.
- Contenido: La idea no es que el usuario navegue en su celular sino que reciba los contenidos que busca.
- Navegar y comprar: En contraposición a lo anterior, el paso del usuario por un portal WAP debe generarle la sensación de búsqueda de información, encuentro con ella y resolución, compra de producto en algunos casos.
- Comercialización: Para que un portal WAP beneficie a una empresa, éste debe tener visibles todos los métodos de comercialización y formas de pago posibles.
- Sencillez: El mensaje debe ser corto, claro y persuasivo, así como los contenidos deben ser fáciles de navegar. **P**

Generá sitios móviles en sólo 5 minutos



- ✓ Panel de control con **tu propia marca**.
- ✓ **Fidelizá** a tus clientes ofreciendo un servicio exclusivo.
- ✓ Generá **más ingresos** sin esfuerzo extra.
- ✓ **Mínima Inversión** inicial y costos competitivos.
- ✓ **Diferenciarte** en el mercado con un producto innovador.

www.dattamovil.com/contumarca

Un producto más de

dattatec.com

Email Marketing con tu marca

Ganá dinero ofreciendo a tus
clientes la mejor herramienta.

