



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO



FACULTAD DE INGENIERÍA

**APLICACIÓN DE LA METODOLOGÍA IWEB EN EL
DESARROLLO DE UN MÓDULO WEB PARA LA BÚSQUEDA
DE CONTENIDOS EN UN SISTEMA EDUCATIVO**

REPORTE DE APLICACIÓN DE CONOCIMIENTOS

**QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN**

**PRESENTA:
VÍCTOR HUGO SOLÍS RAMOS**

**TUTOR:
DRA. ROSA MARÍA VALDOVINOS ROSAS**

Toluca, México, Junio de 2015



UAEM | Universidad Autónoma
del Estado de México

DEPTO. EVALUACIÓN PROFESIONAL

No. Oficio: 0043/2015

Ciudad Universitaria, Toluca, Méx. a 19 de junio del 2015

C. VÍCTOR HUGO SOLÍS RAMOS
PASANTE DE INGENIERÍA EN COMPUTACIÓN
P R E S E N T E

En respuesta a su solicitud, a continuación transcribo el tema aprobado por esta Dirección, que propuso la **DRA. ROSA MARÍA VALDOVINOS ROSAS** con el fin de que lo desarrolle en la modalidad de **REPORTE DE APLICACIÓN DE CONOCIMIENTOS**, le informo que se autoriza la **impresión de su trabajo** para presentar su Evaluación Profesional.

“APLICACIÓN DE LA METODOLOGÍA iWEB EN EL DESARROLLO DE UN MÓDULO WEB PARA LA BÚSQUEDA DE CONTENIDOS EN UN SISTEMA EDUCATIVO”.

| | |
|-------------|--------------------------------------|
| | INDICE GENÉRICO |
| | INTRODUCCIÓN |
| CAPÍTULO 1. | DEFINICIÓN DEL PROBLEMA |
| CAPÍTULO 2. | ANÁLISIS DE ALTERNATIVAS DE SOLUCIÓN |
| CAPÍTULO 3. | IMPLEMENTACIÓN |
| CAPÍTULO 4 | EVALUCIÓN |
| | CONCLUSIONES SUGERENCIAS |
| | REFERENCIAS |

Ruego a usted tomar nota de que en cumplimiento a lo especificado por la Ley de Profesiones, deberá prestar Servicio Social durante un tiempo mínimo de seis meses, como requisito indispensable para sustentar su Evaluación Profesional.

Asimismo, para la elaboración de la **REPORTE DE APLICACIÓN DE CONOCIMIENTOS** y demás trámites, deberá sujetarse a la reglamentación respectiva de esta Universidad.

ATENTAMENTE
PATRIA, CIENCIA Y TRABAJO
“2015, Año del Bicentenario Luctuoso de José María Morelos y Pavón”
FACULTAD DE INGENIERÍA
U.A.E.M.
M. EN I. RAÚL VERA NOGUEZ
DIRECTOR



**/Saha.☺

Cerro de Coatepec S/N, Ciudad Universitaria; Toluca México
Tel. (722) 2-14-08-55 / 2-15-13-51

www.uaemex.mx

AGRADECIMIENTOS

El presente reporte de aplicación de conocimientos fue realizado bajo la supervisión de la Dra. Rosa María Valdovinos Rosas, a quien me gustaría expresar mi más profundo agradecimiento, por hacer posible la realización del trabajo de titulación.

Con todo aprecio a mis padres: **Víctor e Isabel**, por darme la vida, por su ternura, por apoyarme en mi educación universitaria, ya que sin ellos no hubiera logrado mis metas y sueños. Por enseñarme a nunca darme por vencido.

A mi pareja, por sus consejos, paciencia, amor; no hubiese sido posible culminar mis estudios, te agradezco por estar siempre a mi lado.

A mis hermanos, que me impulsan a ser mejor cada día, para que se sientan orgullosos de mí.

A mis amigos, por ser parte de mi vida, de mis momentos tristes y alegres, por apoyarme, por nunca dejarme caer, por estar siempre conmigo.

"Este trabajo fue realizado gracias al apoyo recibido de los proyectos: 3673/2014CE de la UAEM y PEI4-212752 del CONACYT"

Gracias por todo



RESUMEN

Este trabajo muestra las partes que se utilizaron durante el desarrollo de un buscador de contenidos educativos, este módulo web que se incorporó a un sistema educativo fue solicitado por una empresa de servicios de educación en línea, a través a la Facultad de Ingeniería (FI) de la Universidad Autónoma del Estado de México (UAEM) y a la Coordinación de Universidad Abierta y Educación a Distancia (CUAED), de la Universidad Nacional Autónoma de México (UNAM).

Se explican los antecedentes acerca de los buscadores y como son implementados en sistemas para agilizar la obtención de información y que relación tienen con el buscador que se desarrolló.

Durante la construcción del sistema, que solicitó la empresa se establecieron diferentes roles para los involucrados en el sistema, este trabajo se enfoca más en el rol desarrollador, el cual es el que se estableció y se llevó a cabo durante este ciclo del sistema.

Para el desarrollo del módulo de búsqueda se utilizó la metodología iWeb, la cual se centra en los requerimientos del usuario, implementando las diferentes etapas, mismas que se describen desde un punto de vista teórico debido a los acuerdos de confidencialidad firmados con dicha empresa.

Finalmente, se explican los conocimientos aplicados durante este desarrollo a través de conclusiones y se proponen sugerencias para desarrollo futuro.



ÍNDICE GENERAL

| | Pág. |
|---------------------------------------------------------|------|
| Introducción | ix |
| Objetivo general..... | xi |
| Estructura del Reporte | xi |
| Capítulo 1. Definición del problema..... | 12 |
| 1.1 Relación con el plan de estudios cursado | 13 |
| Capítulo 2. Análisis de alternativas de solución..... | 14 |
| 2.1 Buscadores | 14 |
| 2.1.1 Directorios o Índices temáticos..... | 14 |
| 2.1.2 Motores de búsqueda | 16 |
| 2.1.3 Multibuscadores..... | 18 |
| 2.1.4 Metabuscadores..... | 19 |
| 2.1.5 Locales | 20 |
| 2.2 Herramientas de desarrollo dinámico en Web | 22 |
| 2.2.1 Lenguaje de marcado de hipertexto (HTML) | 23 |
| 2.2.2 JavaScript..... | 24 |
| 2.2.3 PHP | 25 |
| 2.3 Base de datos y Servidor Web..... | 27 |
| 2.3.1 Sistema Gestor de Base de Datos PostgreSQL | 27 |
| 2.3.2 Servidor Web Apache | 29 |
| 2.4 Patrones orientados al desarrollo de Software | 30 |
| 2.4.2 Patrón arquitectónico de tres capas | 31 |
| 2.4.3 Patrón de codificación <i>Module</i> | 32 |
| 2.5 Alternativa de solución..... | 33 |
| Capítulo 3. Implementación | 35 |
| 3.2 Formulación y planificación | 35 |
| 3.3 Análisis del buscador | 36 |
| 3.3.1 Análisis de interacción y funcional | 38 |
| 3.3.2 Análisis de la configuración | 42 |



| | |
|-----------------------------------------------------|----|
| 3.4 Ingeniería para el desarrollo del buscador..... | 42 |
| 3.4.1 Diseño arquitectónico | 43 |
| 3.4.2 Diseño de la navegación..... | 44 |
| 3.4.3 Diseño de la interfaz..... | 45 |
| 3.4.4 Diseño del contenido y producción..... | 47 |
| 3.5 Generación de páginas..... | 47 |
| Capítulo 4. Evaluación | 50 |
| 4.1 Pruebas..... | 50 |
| 4.2 Evaluación del cliente..... | 53 |
| Conclusiones y sugerencias..... | 55 |
| Referencias..... | 57 |
| Anexo A - Metodología Iweb..... | 61 |
| Anexo B – Carta de Aceptación | 64 |



ÍNDICE DE FIGURAS

| | Pág. |
|---------------------------------------------------------------------------------------------|------|
| Figura 2.1 Representación de páginas web con vínculos a otras páginas..... | 16 |
| Figura 2.2 Ejemplo de un código en HTML..... | 20 |
| Figura 2.3 Proceso de generación de páginas que utiliza PHP..... | 22 |
| Figura 2.4 Organización de PostgreSQL en su modelo Cliente–Servidor..... | 25 |
| Figura 2.5 Patrón arquitectónico de tres capas utilizado para el desarrollo del buscador... | 28 |
| Figura 3.1 Caso de uso para el buscador de contenidos..... | 36 |
| Figura 3.2 Diagrama de secuencia para el buscador de contenidos..... | 37 |
| Figura 3.3 Diagrama de actividades para el buscador de contenidos..... | 38 |
| Figura 3.4 Estructura jerárquica y navegación..... | 41 |
| Figura 3.5 Diseños de prototipos para la interfaz del buscador..... | 43 |
| Figura 3.6 Componentes utilizados para el patrón arquitectónico de tres capas,..... | 46 |
| Figura A.1 Diagrama del ciclo de vida de la metodología iWeb..... | 63 |



ÍNDICE DE TABLAS

| | Pág. |
|------------------------------------------------------------------------------------------|------|
| Tabla 2.1 Diferencias entre los métodos POST y GET..... | 23 |
| Tabla 2.2 Estadística de Netcraft que muestra los servidores más utilizados..... | 26 |
| Tabla 3.1 Cursos y Talleres impartidos para el desarrollo del módulo de búsqueda..... | 32 |
| Tabla 3.2 Fragmento de actividades, extraídas del plan de trabajo general..... | 33 |
| Tabla 3.3 Fragmento de la matriz de requerimientos utilizada para el análisis..... | 34 |
| Tabla 3.4 Dispositivos móviles utilizados para el desarrollo del buscador..... | 39 |
| Tabla 4.1 Pruebas realizadas el buscador en diferentes dispositivos móviles y ambientes. | 49 |
| Tabla 4.2 Solución a los errores y cambios encontrados en la etapa de pruebas..... | 50 |



INTRODUCCIÓN

Un sistema educativo está dividido por niveles y modalidades, un ejemplo de esto es primaria, secundaria, preparatoria, universidad, etc. y tiene el objetivo de apoyar la educación de una manera uniforme a todos los alumnos con base en lo establecido por las instituciones educativas y las normas que la rigen [16]. Si este sistema educativo se migra a un funcionamiento en línea, con alumnos e instituciones en aulas virtuales, no sería necesario estar de forma presencial para adquirir el conocimiento. Este desarrollo de sistema es complejo ya que no se garantiza que se eduque de una manera uniforme, para esto se hacen uso de los contenidos educativos en línea y evaluaciones. Uno de estos modos de enseñanza en línea es *e-learning*, el cual consiste en el diseño, lanzamiento y evaluación de un curso a través la red, usando el Internet, para educación no presencial, donde los alumnos se encuentran en diferentes lugares y éstos interactúan en diferente tiempo con los recursos informáticos [17].

Una empresa de servicios de educación en línea solicitó a la Facultad de Ingeniería (FI) de la Universidad Autónoma del Estado de México (UAEM) y a la Coordinación de Universidad Abierta y Educación a Distancia (CUAED), de la Universidad Nacional Autónoma de México (UNAM), participar en el desarrollo conjunto de un sistema educativo basado en *e-learning* para reforzar el aprendizaje de estudiantes de preescolar y primaria. El proyecto fue sometido y aprobado en la Convocatoria de Programa de Estímulos a la Innovación 2014, cuyos patrocinadores principales fueron la empresa y la Consejo Nacional de Ciencia y Tecnología (CONACYT).

Cabe mencionar que este sistema educativo propuesto por la empresa ya se había desarrollado una parte en una etapa anterior, y que dicho desarrollo estuvo a cargo del Instituto Tecnológico de Toluca (ITT), este sistema se encontraba incompleto ya que le faltaban muchos módulos web por desarrollar, uno de estos módulos web es el de búsqueda de contenidos educativos.

Para realizar el sistema, se convocó a un grupo de estudiantes de los últimos semestres con el propósito de formar el equipo de desarrollo, buscando entre los candidatos los



perfiles de analista, diseñador, programador, evaluador y documentador. En el desarrollo del sistema, formé parte del grupo de analistas, programadores y diseñadores cuyas funciones fueron:

1. Formulación y planeación del proyecto.
2. Análisis de requisitos del buscador.
3. Ingeniería, para el buscador de contenidos.
4. Generación de páginas del buscador.
5. Pruebas en el buscador de contenidos.
6. Evaluación del cliente.

Como parte de las estrategias de aprendizaje y los requerimientos del usuario, el sistema requería el desarrollo de un módulo para la consulta de contenidos educativos, llamado buscador de contenidos, con el que los alumnos pudieran ejecutar contenidos educativos sin necesidad de estar inscritos en algunos de los existentes. Un buscador de contenido es una herramienta que gestiona bases de datos con distintos contenidos [1]. Algunos ejemplos de estos son los famosos buscadores web que se centran en la búsqueda de páginas como es el caso de Google, Yahoo!, Altavista, entre otros. No obstante, también se puede decir que estos buscadores son considerados como sistemas web que permite a los usuarios encontrar información de algún tema en específico [46]. Esta última definición se acerca más al propósito del desarrollo realizado por centrarse en la recuperación de temas contenidos en una base de datos.

El desarrollo de este módulo fue asignado bajo mi responsabilidad, mismo que describiré en este reporte con el objetivo de que sirva como apoyo para futuros desarrollos, este reporte no compromete la confidencialidad del sistema educativo ya que el reporte de aplicación de conocimientos está aprobado por la empresa, para ser utilizado con motivos de titulación (ver Anexo B).



OBJETIVO GENERAL

Desarrollar un módulo de búsqueda de contenidos en un sistema educativo, que permita la ejecución de contenidos educativos de manera independiente y sin restricciones, con funcionamiento en web y en dispositivos móviles.

ESTRUCTURA DEL REPORTE

El Reporte de Aplicación de Cocimientos, se encuentra estructurado en cuatro capítulos, en el primer capítulo se describe la problemática y los requerimientos que presentó la empresa. El Capítulo dos explica a fondo las soluciones analizadas para el desarrollo del buscador de contenidos educativos. En tanto que el Capítulo tres brinda un panorama del desarrollo del buscador utilizando la metodología iWeb. En el Capítulo cuatro se detallan las pruebas y evaluaciones al buscador de contenidos educativos. Por último, se mencionan algunas conclusiones y recomendaciones.



CAPÍTULO 1. DEFINICIÓN DEL PROBLEMA

Como ya se mencionó en la sección anterior, el sistema tenía un desarrollo inicial en el que no se incluía el módulo de búsqueda de contenidos, entendiendo por contenidos a un curso compuesto por unidades que a su vez se componen por series de ejercicios educativos viables de evaluación. Esta situación constituía un problema debido a que cuando un estudiante deseaba ejecutar un contenido, éste debía estar inscrito en el nivel educativo que lo contiene, situación que restaba flexibilidad de acuerdo al análisis y requerimientos sugeridos por la CUAED.

Derivado de esta situación, se vio la necesidad de desarrollar un módulo con el que se quitara esta restricción, de tal manera que los alumnos pudieran buscar contenidos sin importar el nivel educativo en el que estuvieran dados de alta. En específico, los requerimientos del módulo son:

1. Permitir ingresar cadenas de texto. El usuario podrá introducir letras, números, o símbolos válidos, para posteriormente ser tratados por el módulo de búsqueda.
2. Incluir filtros u opciones de búsqueda. El usuario podrá hacer uso de opciones de búsqueda a través de botones de selección y que el módulo realice una búsqueda precisa.
3. Contar con una interfaz de usuario GUI (del inglés: *Graphical user interface*) amigable o intuitiva con el usuario final. En este contexto, una de las GUI más utilizada es una caja de texto y un botón que realiza la acción. Ejemplo de esto es la interfaz de Google, la cual es fácil de entender para los usuarios [31].
4. Mostrar los resultados de la búsqueda en una lista de contenidos ejecutables. La idea es asegurar que los resultados no se salgan de la GUI y que cada resultado muestre su nombre y descripción así como un vínculo hacia el contenido educativo el cual se ejecutará en pantalla completa.

Otro factor importante a considerar es que la información de los contenidos educativos a buscar se encuentra dispersa en tablas en una base de datos relacional (BDR), limitadas en la búsqueda de acuerdo a la forma en cómo se organiza la BDR y en cómo se obtiene,



ya que muchas bases no cuentan con opciones de búsqueda avanzada de información, si no con la creación de expresiones regulares para búsquedas más complejas.

1.1 RELACIÓN CON EL PLAN DE ESTUDIOS CURSADO

El desarrollo de sistemas es la principal característica que debe cumplir un ingeniero en computación, además de ser una de las líneas de acentuación de la carrera de Ingeniero en computación de la Facultad de Ingeniería de la UAEM, denominada “*Desarrollo de software de aplicación*” en la cual curse mis estudios, por lo que el desarrollo de sistemas, y programación de módulos web son parte de esta línea de acentuación.

Desde mi punto de vista desarrollar el módulo web que solicitó la empresa, es posible gracias a los estudios recibidos en la carrera de Ingeniero en computación, principalmente en las materias de análisis de sistemas, diseño de sistemas, interacción hombre-máquina, entre otras muchas que ayudaron en mi formación como Ingeniero en computación.



CAPÍTULO 2. ANÁLISIS DE ALTERNATIVAS DE SOLUCIÓN

En este capítulo se describe el análisis de diferentes tipos de buscadores, para determinar cuál se adecua más a los requerimientos que solicitó la empresa, así como un análisis de las herramientas utilizadas en la etapa anterior del sistema, tales como lenguajes de programación, la base de datos que aloja la información de los contenidos y los modelos de programación que se siguieron, que en esta etapa son de vital importancia para el desarrollo del buscador. Por último, la metodología de desarrollo iWeb es descrita en el Anexo A.

2.1 BUSCADORES

Actualmente la mayoría de los buscadores tienden a ofrecer el mayor número de servicios disponibles, con lo que sus opciones de búsqueda entre ellos se asemejan cada vez más, lo que dificulta saber que tipo de buscador se va a desarrollar. Sin embargo cada buscador tiene sus propias características y conocer éstas brindará un panorama general para el desarrollo del módulo de búsqueda. A continuación se clasificarán los buscadores por la forma de representar el contenido, la forma de obtenerlo, la combinación de ambos y el lugar en donde se encuentran [1]:

1. Directorios o Índices temáticos
2. Motores de búsqueda
3. Multibuscadores
4. Metabuscadores
5. Locales

2.1.1 DIRECTORIOS O ÍNDICES TEMÁTICOS

Este tipo de buscador es una lista modificada, que ordena los recursos de manera cronológica o geográfica, por temas, categorías, haciéndolo normalmente en forma de árbol



jerárquico [4], con una metodología similar a la de base de datos documentales [5]. Los directorios no contienen los textos completos de los contenidos que enlazan, sólo contienen títulos, descripciones y categoría de temas.

En su funcionamiento, la búsqueda en directorio permite el uso de formularios para realizar consultas directas o para preguntar lo que se está buscando en grandes bases de datos. Estas bases de datos son alimentadas por expertos en el área, que revisan las direcciones que les son enviadas para posteriormente clasificarlas en subdirectorios de forma temática [3]. Algunos de los directorios más utilizados son:

1. Latindex: Sistema Regional de Información en Línea para Revistas Científicas de América Latina, el Caribe, España y Portugal (<http://www.latindex.unam.mx/>).
2. Imbiomed: Índice de Revistas Biomédicas Latinoamericanas (<http://imbiomed.com>).
3. Ulrich: Es el mayor directorio de publicaciones periódicas en el mundo (<http://www.ulrichsweb.com>).
4. Yahoo: Es uno de los buscadores temáticos gratuito más usado en el mundo (<http://dir.yahoo.com>).

Yahoo utiliza la tecnología *WebRank* que utiliza una escala de uno a diez para medir la popularidad de una página web. Sin embargo, *WebRank* no se calcula en función del número de enlaces hacia dicha página (como lo hace Google, ver capítulo 2.1.2), sino que depende de la información obtenida a través de la barra instalada en los navegadores de los usuarios [47].

El algoritmo *WebRank* utiliza los criterios siguientes, para la búsqueda en su directorio [48]:

1. Gran importancia al título de la página a buscar.
2. No es estricto en la repetición de palabras clave (texto a buscar).
3. La densidad de las palabras clave es fundamental para obtener un buen resultado (texto, título, etiquetas meta, etc.).
4. Las palabras clave en la URL tienen un mayor peso, sobre todo cuanto más a la izquierda se encuentren.
5. Los cambios y actualizaciones en Yahoo! son más lentos que en Google



Como ya se mencionó este tipo de buscador realiza una búsqueda por temas, de igual manera que los contenidos educativos que se tienen están clasificados y contenidos en la base de datos, por lo que implementar un índice temático es factible, sin embargo la información no es ingresada de manera directa por personas expertas en el área, sino más bien por el resultado de operaciones de inserción entre el sistema de educación y la base de datos. Además de que la metodología a utilizar es incremental (ver Anexo A) y que implementar este buscador no sería posible debido a lo siguiente: primero los requerimientos no se cumplen totalmente y segundo los requerimientos son volátiles es decir la empresa establece que añadir y quitar.

2.1.2 MOTORES DE BÚSQUEDA

Los motores de búsqueda son programas que realizan las búsquedas dentro de las bases de datos a través de internet, usando palabras clave, los cuales están compuestos básicamente por: un robot, un índice y un mecanismo de búsqueda [6].

1. Robot: Es el programa que examina o rastrea Internet en busca de información sobre las páginas que encuentra, estos robots trabajan de forma particular, simplemente parten de una lista para su rastreo de manera recursiva de los documentos que se referencia.
2. Índice: Base de datos que contiene una copia completa o parcial de los documentos reunidos por el robot.
3. Mecanismo de búsqueda: Es un programa que le permite al usuario encontrar páginas que estén en el índice, según las palabras claves que se establecieron previamente por el usuario.

Google es el motor de búsqueda más conocido y utilizado por millones de usuarios en todo el mundo [7], este cuenta con la tecnología *PageRank* la cual permite que se aprovechen los vínculos (direcciones de páginas web) de manera directa entre distintas páginas, evitando una jerarquía. *PageRank* usa una estructura de vínculos para organizar la información [6], de tal manera que se puede considerar que un vínculo de una página a otra es como una aprobación que la primera página hace a la segunda.



Un ejemplo del mecanismo para construir el motor de búsqueda es el mostrado en la Figura 2.1. Supóngase que la página *A* contiene referencias a las páginas *B* y *C*, la página *B* solo tiene referencia hacia *A*, la página *C* tiene referencias a *B* y *D*, y la página *D* no tiene referencias, de tal manera que la página con más aprobaciones fue la página *B*, la cual tuvo dos aprobaciones por parte de las páginas *A* y *C*, las siguientes páginas con una aprobación fueron *A*, *C* y *D*. Por lo que es considerada importante la página *B* y se le dará mayor ponderación. De esta manera se puede decir que en cuantos más vínculos reciba una página, más popular será [31].

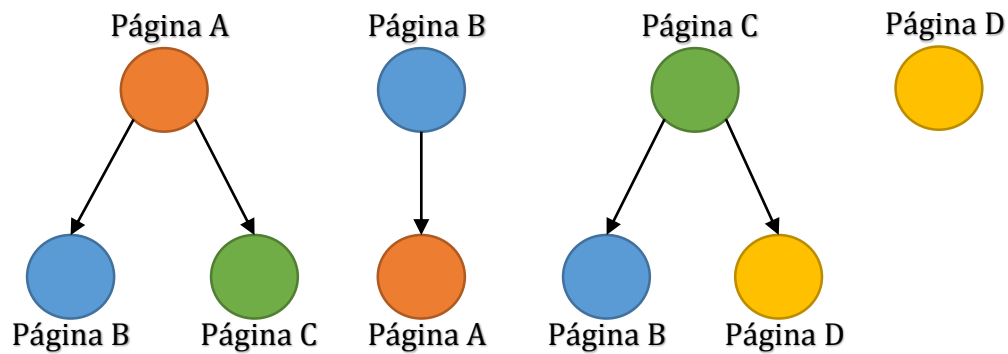


Figura 2.1 Representación de páginas web con vínculos a otras páginas, utilizando la tecnología *PageRank* para determinar cual tiene mayor ponderación [31].

Algunas diferencias de un motor de búsqueda con respecto a uno de índice temático, es que el de índice lo hace de forma manual, utilizando a personas para llenar la base de datos y los motores de búsqueda, hacen uso de un programa para llenar sus bases de datos. Otra diferencia es que los motores de búsqueda pueden crear páginas web dinámicamente y los índices no, finalmente, los motores de búsqueda son exhaustivos y su precisión es deficiente [7].

Uno de los principales beneficios que brinda este tipo de buscador es el uso de programas o robots independientes del mecanismo de búsqueda, para categorizar en este caso los contenidos educativos como ya se mencionó, sin embargo no es necesario saber cuál contenido es mejor y también no se tiene contemplado que el usuario califique los contenidos o que se analice cuáles son los más ejecutados y buscados.



Esta opción podría ser implementada en trabajos futuros si la empresa requiriera que los alumnos consultaran los contenidos más buscados y ejecutados, creando a través de aprobaciones cuales son los contenidos educativos populares. No obstante, por el tiempo disponible para el desarrollo, resultó prohibitiva la implementación para el buscador solicitado (ver Tabla 3.2).

2.1.3 MULTIBUSCADORES

Este buscador es una combinación de los dos anteriores, que son listas de buscadores como motores de búsqueda o índices temáticos, que colocan una lista de los buscadores más comunes y con una caja de texto, que envían a cada uno de ellos la petición de búsqueda [32]. Esto hace que la búsqueda sea híbrida y se obtenga resultados combinados de varios motores o índices de búsqueda, evitando que el usuario tenga que ir a varios buscadores a encontrar la información deseada.

La principal ventaja de estos buscadores es que permiten consultar un gran número de buscadores partiendo de una única página, en tanto que su principal desventaja es que el usuario encontrará un gran número de enlaces y páginas repetidas, este número de resultados repetidos depende de la relación que tengan los buscadores empleados. Por ejemplo, si se tiene un Multibuscador que tenga a Google y Yahoo, regresar 30,000 resultados por ambos de una búsqueda en particular, de los cuales 15,000 podrían ser resultados repetidos, el usuario tendría la mitad de resultados innecesarios.

Un ejemplo de estos buscadores es *Ixquick* (<http://eu.ixquick.com/esp>) que permite buscar en Altavista, *Fast*, Google, MSN, *Open Directory Project* y Yahoo!, al mismo tiempo [5]. Este buscador activa operadores lógicos (AND, OR y NOT), que pueden buscar frases, los resultados recuperados por este Multibuscador van acompañados de un cierto número de estrellas. Cada estrella representa una página en un motor o directorio determinado. Entre mayor sea el número de estrellas que tenga una página esta será relevante, pues cada motor o directorio que contiene el Multibuscador ejecuta su algoritmo propio a la hora de presentar los resultados, y que dicha página aparezca en las primeras pasiones de más de un motor o directorio es signo de su alta penitencia en la búsqueda.



Para fines del buscador solicitado, este tipo de buscador prácticamente no brinda un panorama de lo que se necesita, que es la búsqueda de contenidos educativos ya que es un buscador que depende de otros, por lo que se descarta la utilización de esta alternativa. Pero que también se consideraría para futuros desarrollos que solicite la empresa, específicamente en el uso de una API (del inglés: *Application Programming Interface*), las cuales son funciones que ofrecen una funcionalidad en específica para utilizarla en otro sistema, en este caso la búsqueda [33].

2.1.4 METABUSCADORES

Otro buscador que combina motores e índices temáticos son los Metabuscadores que son un conjunto de programas, de características similares a los agentes autónomos [11]. Éstos son capaces de mostrar la información obtenida de la consulta a otros buscadores de forma automática, obteniendo una combinación de los resultados de cada uno, examinando las páginas visitadas, así como su contenido para obtener un mejor resultado acorde a los criterios de búsqueda del usuario [12]. Unos de los más conocidos y potentes Metabuscadores es MetaCrawler, que incorpora a buscadores como Google, AltaVista, Excite, Infoseek, Yahoo!, Web-Crawler y Lycos, todos al mismo tiempo [34], permitiendo la utilización de operadores booleanos y acotaciones geográficas.

Este tipo de buscador es similar al Multibuscador, con algunas diferencias que se mencionaron como es el uso de agentes, que son entidades (software) que pueden percibir el entorno donde se encuentran, y además actuar sobre él para cumplir su objetivo [35]. Los métodos frecuentemente utilizados para el desarrollo de agentes basados en técnicas de minería de datos (*Datamining*) y aprendizaje para la recuperación de información son:

1. Basados en el contenido: En este enfoque de clasificación de texto, el sistema busca por ítems similares a aquellos que el usuario quiere, basado en la comparación de contenido. El enfoque tiene su origen en Recuperación de la Información, es muy popular en sistemas que trabajan sobre datos de texto como documentos web [49]. Existen problemas concretos cuando se capturan aspectos del contenido, como música, vídeo e imágenes, e incluso para dominios de texto, pues muchas



representaciones capturan solamente ciertos aspectos de dicho contenido. Algunos ejemplos de agentes de este tipo son: WebWatcher, Lira, Musag, Letizia, Personal WebWatcher, CiteSeer, ContactFinder e Internet Fish.

2. Basado en un enfoque colaborativo: En contraste con el enfoque basado en el contenido, que puede ser aplicado a un solo usuario con éxito, el enfoque colaborativo asume que existe un conjunto de usuarios utilizando el sistema. Lo que se ofrece al usuario está basado en la reacción de otros usuarios [50]. El sistema busca por usuarios con intereses similares y recomienda los ítems que considera que un usuario desearía, esto es, analiza la similitud entre usuarios y no entre ítems [1]. Este enfoque se usa, normalmente, para datos no textuales (vídeo o audio, por ejemplo), aunque también se usa para datos de tipo texto, por ejemplo, filtrado de documentos. Está basado en la frecuencia de los temas con que los usuarios acceden a la información, no en el contenido. Algunos ejemplos de agentes que utilizan este enfoque son: Firefly (un agente que funciona en Internet creando un perfil de la persona y hace sugerencias de música y películas basándose en los gustos similares de las personas inscritas <http://www.firefly.com>), Ringo, Siteseer, Phoaks, GroupLense, Referral Web, Fab, WebCobra y Lifestyle Finder.

Para implementar este tipo de buscador se tendrían que analizar estos algoritmos de programación con agentes, además se necesita más tiempo para implementarlo. De igual forma que el anterior este buscador se descarta para implementarlo en el módulo a desarrollar, ya que este tipo de buscadores no se pueden utilizar para búsquedas complejas debido a que cada motor o índice que utiliza trabaja de manera diferente en sus filtros.

2.1.5 LOCALES

Todos los buscadores mencionados anteriormente son buscadores globales, ya que realizan la búsqueda en cualquier país o en toda la Internet, consultando bases de datos en todo el mundo [36]. Ahora bien un buscador local realiza búsquedas limitadas a un ámbito menor o que es lo mismo en bases de datos locales [37]. Este tipo de buscador es común en aplicaciones web o de escritorio, por lo regular su búsqueda es especializada, y están contenidos en un sistema en específico.



Uno de estos buscadores locales es el que se encuentra en el repositorio institucional de la UAEM (<http://ri.uaemex.mx>) el cual solo realiza búsquedas de artículos científicos, tesis, libros, y otros documentos de investigación y no académicos, que están contenidos en la base de datos de la misma Universidad [38].

Con base en lo anterior este tipo de buscador es la opción que se acerca a lo que la empresa solicitó por las siguientes razones:

1. La base de datos que se tiene es local, sin requerir consultas a otras bases a través de internet.
2. El buscador a desarrollar únicamente consultará contenidos educativos por lo que es especializado.
3. Este buscador no necesita uso de algoritmos complejos como los índices temáticos o motores de búsqueda, por tal motivo su desarrollo es ideal para los requerimientos volátiles y el tiempo disponible.
4. Cuenta con características similares a los buscadores de directorios o índices temáticos: se puede categorizar una base de datos relacional en tablas u objetos, si se tiene una tabla que contenga solo un tipo de tema y éste a su vez contenga relaciones hacia otros temas, la búsqueda podría ser a través de palabras clave con opciones de filtrado, a diferencia de los buscadores de directorios que asocian vínculos con palabras clave o patrones.
5. Se pueden utilizar las tablas para categorizar la información y realizar una búsqueda por índices temáticos de forma consistente. En este sentido, es importante mencionar que tipo de gestor de base de datos relacional se utilice ya que dependiendo del gestor se realizarán búsquedas más específicas o menos detalladas.

Para realizar estas búsquedas de manera local se utilizarán sentencias de PostgreSQL especializadas en este ámbito, dichas sentencias permiten obtener palabras complejas a partir de fragmentos de letras que contienen estas palabras, un ejemplo de estas sentencias es el que se muestra a continuación:



1. SELECT *valor-1, valor-2, ...valor-n* FROM *tabla x, tabla y, ...tabla m* WHERE *valor-n* LIKE "%fragmento%";
2. SELECT *valor-1, valor-2, ...valor-n* FROM *tabla x, tabla y, ...tabla m* WHERE (*valor-n* LIKE "%fragmento%" OR *valor-n-1* LIKE "%fragmento%") AND *valor-n-2* = *k*;
3. SELECT *valor-1, valor-2, ...valor-n* FROM *tabla x, tabla y, ...tabla m*
 RIGHT JOIN *tabla-w* AS *a* ON *tabla-w.indice* = *tabla-x.indice*
 LEFT JOIN *tabla-w* AS *b* ON *tabla-y.indice* = *tabla-z.indice*
 WHERE (*valor-n* LIKE "%fragmento%" OR *valor-n-1* LIKE "%fragmento%")
 AND *valor-n-2* = *k*;

En el número uno se tiene una sentencia básica para la consulta ya que obtiene n valores de m tablas posibles, utilizando la función LIKE para buscar fragmentos de palabras en dichos campos de las tablas. En el ejemplo dos, se tiene una consulta más compleja que el uno, y hace la combinación con operadores lógicos OR y AND, que unen un conjunto de resultados relacionados. Para el ejemplo tres, la consulta es más especializada y regresa datos más precisos ya que hace uso de combinaciones de tablas con JOIN. La sentencia LEFT encuentra registros de la tabla izquierda que se correspondan con los registros de la tabla derecha y si un valor de la tabla izquierda no se encuentra en la tabla derecha, el registro muestra los campos correspondientes a la tabla de la derecha de forma nula. La sentencia RIGHT opera del mismo modo sólo que la tabla derecha es la que localiza los registros en la tabla izquierda [51].

2.2 HERRAMIENTAS DE DESARROLLO DINÁMICO EN WEB

El sistema educativo que la empresa requería, ya contaba con algunos módulos desarrollados, los cuales estaban realizados en su totalidad en HTML, con JavaScript para la parte de la GUI, y para la parte de lógica de programación PHP. Dado que se trata de dar continuidad al desarrollo ya existente, las siguientes secciones proporcionan una descripción de cada una de éstas y un análisis de las mismas con motivo de que sirvan para el desarrollo planteado.



2.2.1 LENGUAJE DE MARCADO DE HIPERTEXTO (HTML)

HTML es un lenguaje de marcas de texto utilizado normalmente en web para la estructura de documentos. La mayoría de los documentos HTML tienen estructuras comunes (títulos, párrafos, listas, entre otras) que son definidas por este lenguaje mediante etiquetas (Método para pasar parámetros a subrutinas), por lo que todo aquello que no sea una etiqueta es parte del documento mismo [39]. Este lenguaje no describe la apariencia del diseño del sistema, sólo ofrece a cada plataforma que le dé formato según su capacidad y la de su navegador (tamaño de la pantalla, fuentes que tiene instaladas).

Algunas de las ventajas de HTML son [20]: compatibilidad con la mayoría de los navegadores, multiplataforma, facilidad de aprendizaje (debido al reducido número de etiquetas que usa), factibilidad de programación para dispositivos móviles, computación en la nube y funcionamiento en red, que lo hacen conveniente para el proyecto que se desarrolla. Por otro lado, como desventaja tiene que los programas de texto son poco amigables y tienen una interfaz restringida [28]. La Figura 2.2 muestra un fragmento de código realizado en HTML en él se puede apreciar las etiquetas comunes los atributos que la conforman.

```

1 <html>
2   <head>
3     <title> nuevo documento </title>
4   </head>
5   <body bgcolor="white" text="black">
6     <h2> Nuevo Doc </h2>
7     por <a href="/"> Victor Solis </a>, revisado 02 de junio de 2015
8     <hr>
9     texto introductorio
10    En primer lugar <h3> Subtitulo </h3>
11    más texto
12    <h3> Segundo subtitulo </h3>
13    concluir el texto
14    <hr>
15    <a href="mailto:vhsr@uaem.mx">
16      <address> vhsr@uaem.mx </address>
17    </a>
18  </body>
19 </html>

```

Figura 2.2. Ejemplo de código en HTML, se muestran las etiquetas y sus atributos [29].



De la Figura anterior, en la línea 1 se muestra la etiqueta de apertura de documento `<html>`, en la línea 2 la etiqueta para la apertura del encabezado `<head>`, que contiene la etiqueta `<title>` para dar nombre al documento. En la línea 5 se visualiza la etiqueta `<body>`, que contiene el cuerpo del documento, donde va todo el diseño para la UI, esta etiqueta tiene los atributos `bgcolor` que sirve para seleccionar el color de fondo y el atributo `text` para el color del texto. En la línea 10 y 6 son subtítulos de diferentes tamaños `<h2>` es mayor que el subtítulo `<h3>`. En la línea 15 la etiqueta `<a>` sirve para realizar vínculos o referencias con su atributo `href` que contiene la dirección del vínculo. Por ultimo todas las etiquetas deben tener su etiqueta de cierre, en la línea 18 y 19 se muestran la etiqueta de cierre de cuerpo del documento y la etiqueta de cierre de todo el documento.

Derivado de lo anterior, la compatibilidad con otros lenguajes y su facilidad de uso lo hace ideal para el desarrollo de las GUI de buscador de contenidos, por las siguientes razones: en primera es el lenguaje de marcado estándar para el desarrollo web, también varios módulos con los que cuenta el sistema de la empresa así como los contenidos educativos ejecutables están desarrollados en HTML.

2.2.2 JAVASCRIPT

JavaScript es un lenguaje interpretado semejante a Visual Basic, Perl, TCL que se ejecuta del lado del cliente eficaz para trabajar en Web debido a su compatibilidad con HTML y CSS [19], que permite enviar documentos a través de la Web que incorporan el código fuente de un programa, convirtiéndose de esta forma en documentos dinámicos en lugar de simples fuentes de información estáticas.

Los programas en JavaScript no son la primera forma que se conoce para transformar información, dado que el uso de CGI (del inglés: *Common Gateway Interface*) está ampliamente difundido. La diferencia básica que existe entre un programa CGI y uno escrito en Java Script es que el CGI se ejecuta en el servidor de páginas Web mientras que el programa en Java script se ejecuta en el cliente, (es decir, en el navegador, [30]).

Al ser JavaScript un lenguaje orientado a objetos y orientado a eventos, implica que gran parte de la programación se centra en describir objetos (con sus variables de instancia y



métodos de clase) y escribir funciones que respondan a movimientos del ratón, pulsación de teclas, apertura y cerrado de ventanas o carga de una página, entre otros, que lo hacen atractivo para la programación web.

De manera similar que HTML, JavaScript está contenido en la mayoría de los módulos web ya desarrollados del sistema educativo, lo que da la ventaja de seguir bajo el mismo esquema de diseño dinámico de las GUI. Los contenidos educativos y el mecanismo de evaluación que la empresa desarrolla también están realizados en este lenguaje, por lo que para que éstos puedan ser ejecutados de manera independiente es necesario crear nuevas funciones en este lenguaje para adaptarlos a esta funcionalidad. Por lo tanto esta opción que propone la empresa será utilizada para el desarrollo del buscador de contenidos educativos.

2.2.3 PHP

PHP es un lenguaje de programación que se ejecuta de lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse, la mayor parte de su sintaxis ha sido tomada de C, Java y Perl [10]. Para su funcionamiento se necesita tener instalado el servidor aplicación Apache. La Figura 2.3 muestra el proceso de generación de páginas HTML, cuando un cliente a través de su navegador realiza una petición al servidor que contiene código en PHP.

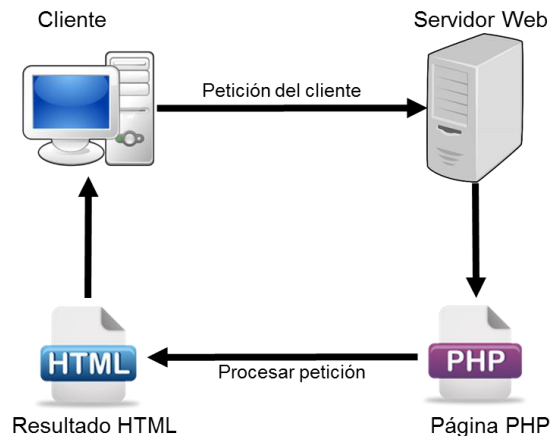


Figura 2.3 Proceso de generación de páginas que utiliza PHP, para generar código HTML en respuesta a la petición de un cliente [39].



Como se muestra en la Figura anterior, este proceso se lleva a cabo a través de una arquitectura cliente-servidor, en donde el cliente a través de su navegador web realiza una petición hacia el servidor el cual contiene la página PHP que atiende dicha solicitud. Los métodos utilizados por el cliente para enviar datos son GET y POST, en donde la función GET envía los datos por la URL del navegador y esto son visibles por el usuario, y el método POST los envía a través de cuerpo del mensaje el cual no es visible por los usuarios de la web [45]. Para apreciar mejor las diferencias de estos dos métodos la Tabla 2.1 muestra las diferencias entre algunos conceptos y observaciones.

Tabla 2.1 Diferencias entre los métodos POST y GET, utilizados para el envío de mensajes con páginas HTML y PHP [45].

| Método | Concepto | Observaciones |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| GET | Este método envía los datos de forma visible, a través de la cabecera de la petición utilizando la URL del navegador y cualquier persona los puede ver. | Ejemplo de datos visibles para el usuario: <code>http://www.uaemex.mx/pagina.php?nombre=victor&apellido=solis</code> En este ejemplo los datos nombre y apellido son visibles y accesibles por los usuarios para consulta o modificación |
| POST | Este método envía los datos de forma oculta, enviados a través del cuerpo de la petición utilizando formularios para su envío. | La única forma de usar el método POST es usando formularios en HTML como el siguiente: <code><form action="http://www.uaemex.mx/pagina.php" method="post"></code> |

La compatibilidad de PHP con HTML y JavaScript no es la principal razón por la que se utiliza para desarrollar el buscador de contenidos educativos, el modelo de programación utilizado, que divide la aplicación en módulos o funciones (ver capítulo 2.4.3) es otro de los aspectos que lo hacen viable de uso. De este modo, las funciones nuevas que se incorporen seguirán este modelo de programación para que en futuros desarrollos se utilicen.



2.3 BASE DE DATOS Y SERVIDOR WEB

Una base de datos es un conjunto de información organizada de forma estructurada, que está agrupada gracias a que contienen al menos una característica en común [13]. Éstas permiten recopilar información para su posterior consulta, las cuales tiene en su mayoría un motor de búsqueda interno, con características especiales que permiten de manera rápida y eficaz la ubicación de cada tema o dato [4]. Derivado de esto la base de datos que propone la empresa es la que ofrece el Sistema Gestor de Base de Datos (SGBD) de PostgreSQL, que es una base de datos objeto relacional, la cual permite establecer relaciones entre los datos (que están guardados en objetos llamados tablas), y relacionar los datos de ambas tablas, a esta forma de relación se le denomina Modelo Relacional [15].

2.3.1 SISTEMA GESTOR DE BASE DE DATOS POSTGRESQL

Un SGBD es un conjunto de programas que actúan de interfaz entre los datos y las aplicaciones, este crea y organiza toda la base de datos y atiende todas las solicitudes de acceso hechas a la base de datos tanto de los usuarios como de las aplicaciones [21]. EL SGBD de PostgreSQL, está bajo licencia BSD que nos permite ver el código y modificarlo libremente, con la opción de cerrar la aplicación para beneficio propio [40]. Este utiliza un modelo cliente-servidor, el cual usa multiprocesos en lugar de multihilos esto garantiza la estabilidad del sistema, ya que un fallo en uno proceso no afecta el resto y el sistema continuará funcionando. En la Figura 2.4 se muestra de manera general los componentes importantes del sistema gestor PostgreSQL que se describen a continuación [22]:

1. Aplicación cliente: Esta es la aplicación cliente que utiliza PostgreSQL como administrador de bases de datos. La conexión puede ocurrir por TCP/IP ó sockets locales.
2. Demonio *postmaster*: Este es el proceso principal encargado de escuchar por un puerto/socket por conexiones entrantes de clientes. También es el encargado de crear los procesos hijos que se encargarán de autenticar estas peticiones, gestionar las consultas y mandar los resultados a las aplicaciones clientes.
3. Ficheros de configuración: Los tres ficheros principales de configuración utilizados son `postgresql.conf`, `pg_hba.conf` y `pg_ident.conf`.



4. Procesos hijos postgres: Procesos hijos que se encargan de autenticar a los clientes, de gestionar las consultas y mandar los resultados a las aplicaciones clientes.
5. PostgreSQL *share buffer cache*: Memoria compartida usada para almacenar datos en caché.
6. *Write-Ahead Log (WAL)*: Encargado de asegurar la integridad de los datos, usando herramientas de recuperación.
7. *Kernel disk buffer cache*: Caché de disco del sistema operativo.
8. Disco: Es donde se almacenan los datos y toda la información.

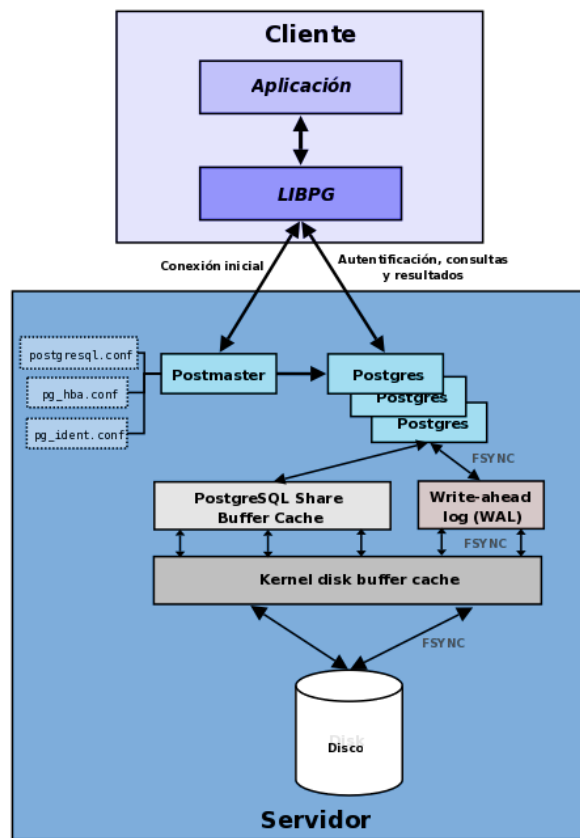


Figura 2.4 Organización de PostgreSQL en su modelo Cliente-Servidor [22]

Este SGBD usa sentencias SQL, para la definición, manipulación y control de la base de datos relacional [14]. Por tal motivo este SGBD es la alternativa única, ya que las sentencias utilizadas por otros módulos se basan en este lenguaje, además de que todas las tablas y modelo relacional el cual por motivos de confidencialidad no es posible describirlo en este



reporte, están en esta base de datos. Esto hace que se reutilicen tablas así como procedimientos que estuvieran almacenados en la base de datos para futuros desarrollos.

Una de las principales características por las cuales este SGBD, es adecuado para el desarrollo del buscador es que ofrece sentencias para la manipulación de los datos utilizando las sentencias (ver capítulo 2.1.5) como son SELECT, que es utilizada para obtener datos según el criterio que se establezca, DISTINCT que elimina resultados repetidos, y la más importante LIKE, que prácticamente recibe una cadena de texto y encuentra datos que contengan dicha cadena[41], lo cual cuando se desarrolle el buscador de contenidos educativos se necesitarán de todas estas sentencias para garantizar el cumplimiento los requerimientos solicitados anteriormente.

Es importante aplicar la normalización a las tablas de la base de datos ya que para realizar búsquedas con relaciones sobre tablas necesitamos que los datos estén concisos, para así obtener un resultado esperado.

2.3.2 SERVIDOR WEB APACHE

El servidor web utilizado por el sistema educativo es Apache que es la plataforma web número uno en el mundo, de código abierto utilizado por empresas, con un diseño modular que permite ampliar su funcionalidad creando módulos en C o Perl, también tiene soporte para PHP y que es un servidor multiplataforma [18]. La compañía Netcraft que es una compañía dedicada a encuestas, publica periódicamente estadísticas de los servidores web más utilizados. La Tabla 2.2 muestra las estadísticas publicadas del mes de mayo y abril del presente año.

Tabla 2.2 Estadística de Netcraft que muestra los servidores más utilizados [42].

| Servidor | Abril 2015 | Porcentaje | Mayo 2015 | Porcentaje |
|-----------|-------------|------------|-------------|------------|
| Apache | 333,285,741 | 39.25% | 336,813,959 | 39.26% |
| Microsoft | 236,288,843 | 27.83% | 247,784,668 | 28.88% |
| nginx | 126,274,778 | 14.87% | 123,697,645 | 14.42% |
| Google | 20,051,433 | 2.36% | 20,103,068 | 2.34% |



De la tabla anterior se puede visualizar que Apache ocupa el primer lugar de servidores instalados, lo que lo hace confiable para el desarrollo de buscador de contenidos educativos, además de brindar las siguientes características: la primera que tiene una arquitectura modular en la cual las peticiones por parte de los clientes son atendidas por el módulo especializado, es el caso del módulo que se encarga de interpretar las páginas PHP [43], que para el desarrollo del buscador es necesario ya que como se mencionó anteriormente el sistema se desarrollará por completo en PHP.

Los servidores de la FI que se utilizaron para el desarrollo del módulo de búsqueda tiene instalado Apache 2.3 en sistema operativo Linux, lo que es una razón conveniente para utilizar este servidor. Además otra razón importante a considerar es que la empresa tiene sus propios servidores con sistema operativo Red Hat Linux 6 con Apache, en los cuales se instalará el sistema educativo cuando se termine esta etapa de desarrollo.

2.4 PATRONES ORIENTADOS AL DESARROLLO DE SOFTWARE

Una herramienta fundamental para el desarrollo del buscador es el uso de patrones basados en el desarrollo de software, la empresa contaba ya con algunos de estos patrones para la construcción de módulos anteriores. Se analizarán estos patrones para determinar si son factibles en el desarrollo del buscador. Primero los patrones de software son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de sistemas y otros ámbitos referentes al diseño de interacción o interfaces [24]. Para que una solución sea considerada un patrón debe poseer ciertas características: haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores y ser reutilizable, lo que significa que es aplicable a diferentes problemas de software en distintas circunstancias [44]. Los patrones de software pueden clasificarse según el nivel de abstracción [25]:

1. Patrones arquitectónicos. Centrados en la arquitectura del sistema, que definen una estructura fundamental sobre la organización del sistema.
2. Patrones de diseño. Esquemas para refinar los componentes de un sistema, o sus relaciones. Describen una estructura recurrente y común de componentes comunicantes que resuelven un problema de diseño dentro de un contexto.

3. Patrones de codificación o modismos (*idioms*). Son patrones que ayudan a implementar aspectos particulares del diseño en un lenguaje de programación específico.

De los patrones mencionados anteriormente la empresa propone utilizar el patrón arquitectónico de tres capas para separar la programación en diferentes capas y el patrón de codificación module, para seguir el esquema de programación que se ha venido trabajando en la etapa anterior del sistema.

2.4.2 PATRÓN ARQUITECTÓNICO DE TRES CAPAS

Este patrón de software permite dar una visión de toda la distribución de los componentes de la aplicación separándola en la capa de presentación, la capa de negocios y la de datos [26] (ver Figura 2.5).

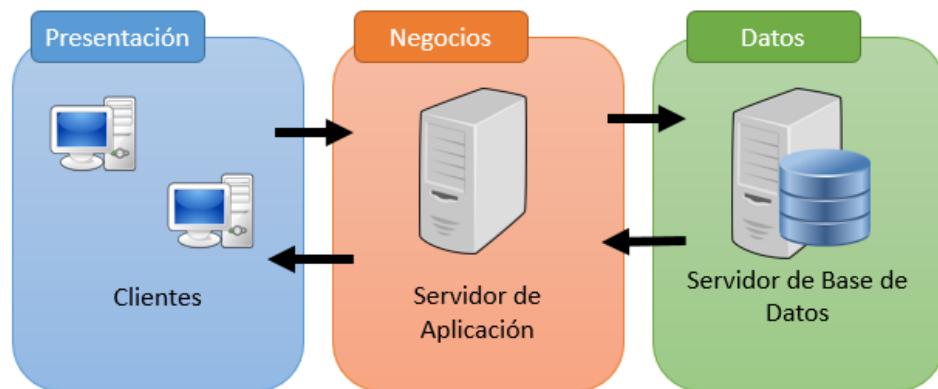


Figura 2.5 Patrón arquitectónico de tres capas utilizado para el desarrollo del buscador [27].

De la Figura 2.5 se ilustra a los componentes de la arquitectura de tres capas, este tipo de modelo es lineal, es decir para pasar de una capa otra se tiene que pasar por la capa anterior, estas capas tienen las siguientes características:



1. Presentación: Comprende toda la GUI para manejar la interacción entre el usuario y la aplicación. Interfaces en HTML que ejecutan código en JavaScript que el buscador contendrá, así como algunas plantillas en PHP.
2. Negocios: Contiene toda la lógica de la aplicación es decir la funcionalidad específica que la aplicación debe hacer. Las funciones sobre la base de datos y la validación de los datos. Para el buscador esta capa contendrá todas las funciones nuevas para el buscador así como las ya realizadas en PHP.
3. Datos: Tiene que ver con la comunicación con otros sistemas que llevan a cabo tareas en nombre de la aplicación. Monitores de transacciones, base de datos, sistemas de mensajería, etc. En este caso el único sistema con el que se comunica la aplicación es el del SGBD de PostgreSQL.

La ventaja de este patrón arquitectónico es que permite agrupar los componentes del sistema, en diferentes unidades lógicas funcionales durante las etapas de la metodología. De esta forma la propia arquitectura, determina la distribución de los componentes, y su agrupación en capas lógicas [27]. Por tal motivo este patrón fue propuesto por la empresa, además de que en la etapa anterior del sistema se utilizó este patrón de software.

Otra razón es que se adapta a la metodología iWeb utilizada para este desarrollo, permitiendo tener un correcto diseño para el buscador solicitado por la empresa. En la etapa de Generación de páginas de la metodología iWeb (ver Anexo A), es donde se pretende utilizar este patrón de software, ya que es una etapa en donde se realiza toda la programación de los componentes del sistema, y que es por esta razón que esta metodología se adapta al patrón descrito.

2.4.3 PATRÓN DE CODIFICACIÓN *MODULE*

Una de las características fundamentales es que el sistema educativo estaba basado en módulos, donde cada módulo tiene una función específica, lo que se pretende para el desarrollo del buscador es realizar o seguir este patrón de software, para que la estructura del sistema no se pierda. Para esto se utilizará el patrón *module* que es un patrón utilizado para implementar el concepto de módulos de software, proporcionando características y estructuras sintácticas, definidas por el paradigma de Programación Modular a Lenguajes



de Programación, que no lo soportan completamente, o que solo lo soportan de forma incompleta [24]. Aunque PHP es un lenguaje que soporta funciones, también soporta otras estructuras como son clases, scripts, etc., que lo hacen un lenguaje mixto [45]. De manera que únicamente se utilice PHP como un lenguaje modular.

Este patrón permite llevar a cabo la reutilización de funciones en código en PHP, el sistema educativo cuenta con módulos que realizan funciones similares a las requeridas por el buscador en la parte de la ejecución de los contenidos, permitiendo reutilizarlas, además de que las nuevas funciones creadas seguirán el mismo patrón de diseño para que en futuros desarrollos se reutilicen. Las ventajas y desventajas que ofrece este patrón de codificación son [44]:

1. Ventajas

- Al aplicar el patrón *module*, un problema complejo debe ser dividido en varios subproblemas más simples, y estos a su vez en otros subproblemas más simples.
- En caso de que un módulo necesite de otro, puede comunicarse con éste mediante una interfaz de comunicación que también debe estar bien definida.
- Es fácil de mantener y modificar
- Es más fácil de escribir y depurar
- Facilidad de controlar es decir descompone un problema en estructuras jerárquicas, de modo que se puede considerar cada estructura desde dos puntos de vista

2. Desventajas

- No se dispone de algoritmos formales de modularidad, por lo que a veces los programadores no tienen claras las ideas de los módulos
- La programación modular requiere más memoria y tiempo de ejecución

2.5 ALTERNATIVA DE SOLUCIÓN

De las alternativas que se mencionaron anteriormente sobre el tipo de buscador a implementar, las herramientas de desarrollo dinámico e web, los servidores de base de datos y web, así como los patrones de software se determinó lo siguiente: el buscador que más se adecua a las necesidades demandadas por la empresa es un buscador local. Esto debido a las razones ya mencionadas como son el uso de una base local y su fácil desarrollo



que garantiza su construcción derivado de los tiempos limitados que se cuenta para su entrega. Utilizando los lenguajes HTML, JavaScript Y PHP, junto con la base de datos PostgreSQL y los patrones de software de tres capas y *module*, para el desarrollo del mismo.

Este desarrollo se realizará siguiendo cada una de las etapas de la metodología iWeb, en donde en cada etapa se describirá de manera general lo que se realizó, esto debido a los acuerdos de confidencialidad llevados a cabo por la empresa. Cabe destacar que esta metodología ya se había utilizado en una etapa anterior al sistema, por lo que fue la primera propuesta para el desarrollo. En el siguiente capítulo se describirá la forma como se realizó el desarrollo del buscador, con base en las etapas de esta metodología.



CAPÍTULO 3. IMPLEMENTACIÓN

Para el desarrollo del buscador de contenidos educativos se aplicará la metodología iWeb en cada una de sus etapas de desarrollo, describiendo lo que se realizó en cada una de ellas, también haciendo uso de todas las herramientas descritas en el capítulo anterior como son el uso del lenguaje PHP, la base de datos, servidor web, patrones de software, etc., para el desarrollo del módulo.

3.2 FORMULACIÓN Y PLANIFICACIÓN

Las primeras dos etapas de esta metodología son la formulación y planificación en donde el equipo de desarrollo de la FI identificó las metas y objetivos del sistema que la empresa solicitó. Para esto se realizaron una serie reuniones entre la FI y la empresa, mismo periodo en el cual se impartieron cursos de capacitación e inducción, para conocer más el sistema (ver Tabla 3.1).

Tabla 3.1 Cursos y Talleres impartidos para el desarrollo del módulo de búsqueda.

| Curso o Taller | Descripción |
|----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Curso de inducción a al sistema | Descripción del sistema desarrollado y la expectativa de desarrollo. |
| Manejo de bases de datos en PostgreSQL | El objetivo del taller fue para actualizar y obtener los conocimientos necesarios para el manejo de PostgreSQL. |
| Programación con HTML5 | Se participó en este taller para adquirir los fundamentos para programar en HTML. |
| Programación en PHP | Se participó en este taller para actualizar, practicar y adquirir conocimiento más a fondo de la programación utilizada para el desarrollo del sistema. |
| Taller de Metodología SW | Este taller fue realizado por parte de personal de la empresa para darnos un panorama general del desarrollo de sistemas utilizando una metodología MAGTIC. |



Los primeros cuatro cursos de la Tabla 3.1 se brindó un panorama general de lo que trataría el sistema a desarrollar, así como un reforzar los conocimientos para el desarrollo de los módulos que se asignaron a los becarios, el último taller que es el de la metodología MAGTIC la cual por motivos de confidencialidad no es posible describirla, esta metodología fue utilizada para la documentación que la empresa requeriría en el desarrollo del sistema.

Derivado de las reuniones, se obtuvieron acuerdos por parte del equipo de desarrollo de la FI y la empresa para estimar el costo del desarrollo del sistema, su factibilidad de desarrollo (ya que se trataba de un sistema truncado al que le faltaban módulos por desarrollar), riesgos potenciales y cuanto esfuerzo de desarrollo se requería para cada incremento en la funcionalidad, creando así un plan de trabajo. El fragmento de actividades, extraídas del plan de trabajo general, correspondientes al diseño del buscador se muestran en la Tabla 3.2.

Tabla 3.2 Fragmento de actividades, extraídas del plan de trabajo general.

| Actividad | Duración | Perfil |
|--------------------------------------------------|----------|------------------------------|
| Desarrollo de interfaces | 13 días | Líder Desarrollo SW UAEM 1T1 |
| Actualización de interfaces | 3 días | Líder Desarrollo SW UAEM 1T1 |
| Desarrollo del buscador de contenidos educativos | 20 días | Líder Desarrollo SW UAEM 1T1 |

De acuerdo con que se ilustra en la Tabla 3.2 el tiempo que se propuso para el desarrollo del módulo fue de 36 días, sin embargo el tiempo se extendió debido a que la empresa solicito cambios y correcciones al buscador llevando así más tiempo para su desarrollo. Otro factor es que no fue el único módulo que se desarrollaba y que estaba bajo mi responsabilidad el cumplir con las fechas señaladas bajo el perfil “Líder Desarrollo SW UAEM 1T1”.

3.3 ANÁLISIS DEL BUSCADOR

De acuerdo a las etapas de Formulación y Planificación, se realizaron reuniones para determinar los requerimientos funcionales y no funcionales. El equipo de desarrollo de la FI y pedagogos de CUAED en conjunto con la empresa discutieron los elementos del



contenido que se incorporarían al sistema, en esta etapa se participó, haciendo uso de herramientas como matrices de requerimientos y minutas siguiendo la metodología de documentación MAGTIC, que facilitaron la obtención detallada de los elementos que contendría el sistema educativo, en especial el módulo de búsqueda. Para el levantamiento de requerimientos se realizó utilizando el formato propuesto por la empresa ilustrado en la Tabla 3.3, de los cuales no son los requerimientos exactos pero se basa de manera general en lo que la empresa solicitó.

Tabla 3.3 Fragmento de la matriz de requerimientos utilizada para el análisis.

| Identificador | Actividad | Descripción | Comentarios |
|---------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IDEN_001 | Búsqueda | Se permitirá el ingreso caracteres. | Se utilizará una caja de texto. |
| IDEN_002 | Ejecutar Contenidos | El alumno podrá consultar contenidos y ejecutarlos de forma individual, sin necesidad de que pertenezcan a un curso asignado. | El avance de los contenidos se registra en la base de datos y genera logros que serán visualizados en una sección específica por el usuario. |
| IDEN_003 | Resultados de la búsqueda | Los resultados se muestran en una lista de contenidos con un vínculo al contenido. | El filtro y la visualización son similares a como se consultan los contenidos en otros módulos. Se aprovechará toda el área del contenido de la página. |
| IDEN_004 | Filtros de búsqueda | Los resultados pueden incluir diferentes áreas de conocimiento. | El buscador funciona específicamente para los contenidos individuales. |

Los campos incluidos en la Tabla 3.3 corresponden a lo siguiente:

1. Identificador. Es donde se enumeran los requerimientos siguiendo el formato en establecido por la metodología MAGTIC.
2. Actividad. Corresponde a las actividades a realizar para el desarrollo del buscador.



3. Descripción. Describe de manera detallada el requerimiento.
4. Comentarios. Es para los comentarios, aquí se escribieron actividades que el buscador podrá o debería realizar.

Durante el análisis se determinaron los aspectos de la interfaz de usuario, las cuales ya tenían un modelo establecido, con diseños propios de la empresa y los pedagogos los cuales establecieron diferentes colores en la interfaz, según el nivel educativo o el tipo de usuario. Para este desarrollo se determinaron los perfiles de alumno, tutor, padre de familia y administrador, de los cuales el perfil de alumno se dividía en preescolar, primaria I, primaria II y primaria III, el desarrollo de este buscador de contenidos educativos debería ser funcional para todos los perfiles de alumno.

3.3.1 ANÁLISIS DE INTERACCIÓN Y FUNCIONAL

Para este análisis de interacción se utilizaron diagramas de UML, donde se detallaron los casos de uso para determinar la interacción del usuario con el sistema. Esta interacción pretende identificar los componentes que se utilizaran para el desarrollo del buscador de contenidos educativos. Como es el caso de la inserción de texto, el uso de filtros los cuales se amplían debido a que se validan de forma que para el usuario solo le muestre los filtros que tengan relación y por último la actividad de realizar la búsqueda.

El caso de uso general para el buscador es el que se muestra en la Figura 3.1, el cual consta de dos actores principales que son el alumno en sus diferentes perfiles (preescolar, primaria I, II y III) y el actor del módulo de búsqueda quien atenderá las solicitudes del alumno. De igual modo, consta de cuatro casos:

1. Ingresar texto describe como el usuario puede a través de la caja de texto insertar cadenas de caracteres y ser utilizadas para la búsqueda de contenidos.
2. Selección de filtros, en éste el usuario puede seleccionar diferentes opciones para especializar la búsqueda,
3. Revisar resultado de elegir una opción, si ésta tiene relación con la opción siguiente entonces se realiza un acotamiento de las opciones elegidas.

4. Realizar la búsqueda la cual está relacionada con el módulo de búsqueda de contenidos.

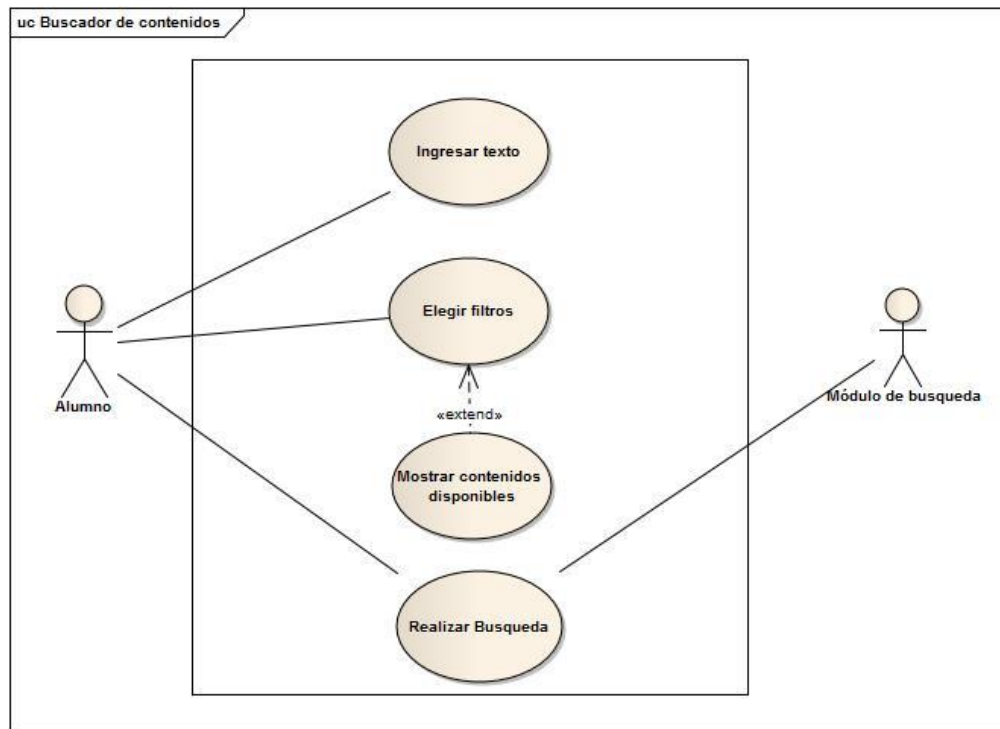


Figura 3.1 Caso de uso para el buscador de contenidos.

Para el análisis funcional se utilizaron los diagramas de secuencia y de actividades, los cuales permitieron tener un panorama detallado de los requerimientos del usuario. Para el diagrama de secuencia de la Figura 3.2 se detalla como el usuario inicia la solicitud a la interfaz y posteriormente la interfaz realiza una petición a las funciones del buscador que se encuentran en la capa de negocios, las cuales regresan una lista de contenidos educativos que son mostrados en GUI al usuario con un vínculo para su ejecución. También este diagrama muestra los parámetros que son enviados de la GUI a las funciones del buscador, que son el conjunto de opciones de los filtros que el usuario seleccionó previamente y que este conjunto de opciones son analizados por el módulo de búsqueda para crear la sentencia SQL de forma dinámica.

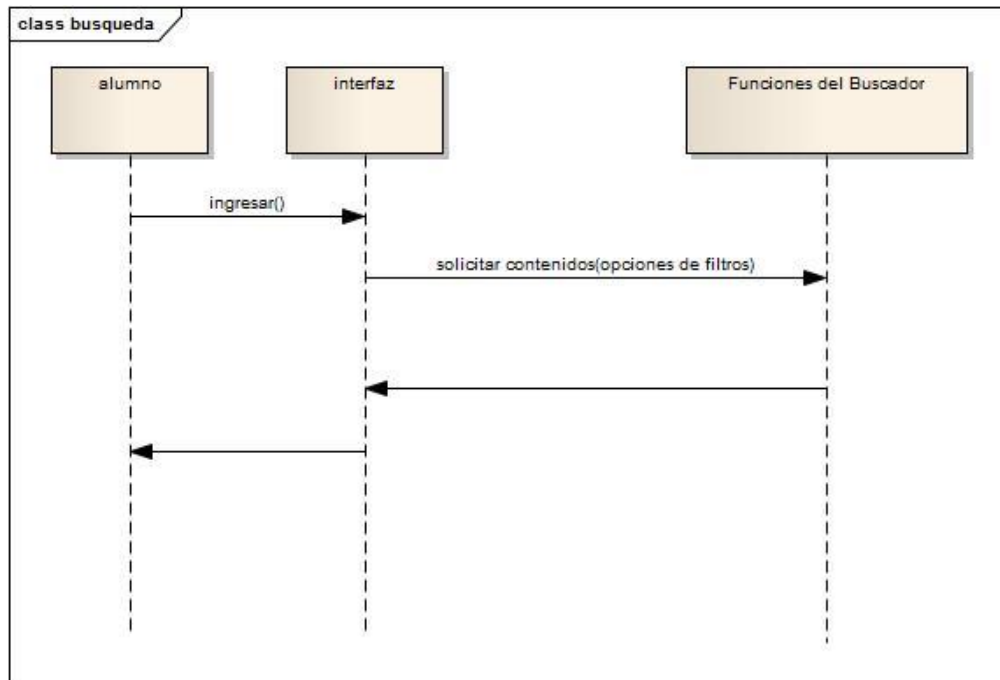


Figura 3.2 Diagrama de secuencia para el buscador de contenidos.

El siguiente diagrama para el análisis funcional es el de actividades que muestra acciones a realizar por cada actor, el diagrama de la Figura 3.3 ilustra las actividades que se realizan para crear una búsqueda entre el usuario del sistema y el módulo de búsqueda, este empieza cuando el usuario selecciona un filtro de búsqueda los cuales deben estar relacionados, el usuario ingresa la cadena a buscar, el módulo de búsqueda analiza los parámetros enviados y regresa una lista de contenidos en la GUI y si son los contenidos deseados entonces se pueden ejecutar por parte del usuario.

Una característica importante es que la actividad de desplegar contenidos educativos que se muestra en la Figura 3.3, es parte de otro módulo del sistema educativo y que con adecuaciones se adaptado para ser utilizado en el buscador.

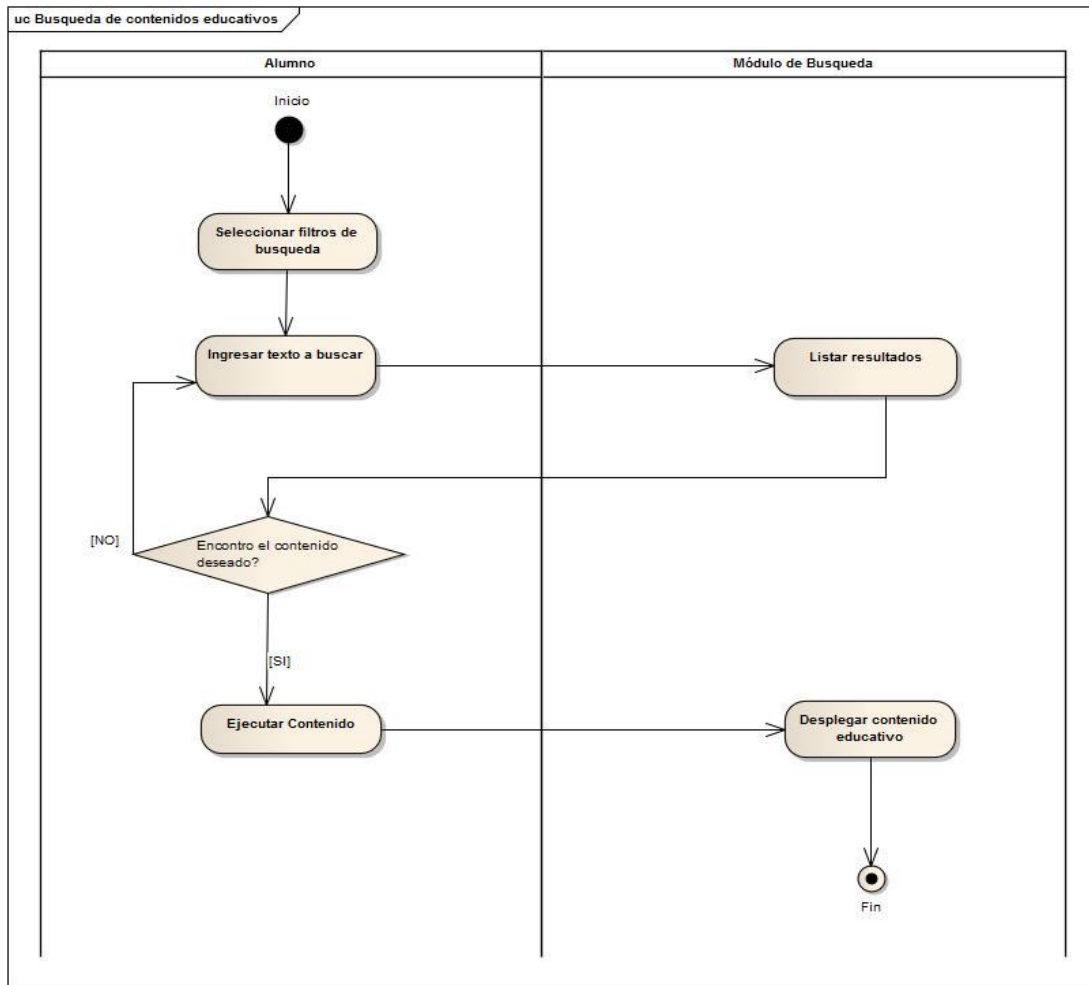


Figura 3.3 Diagrama de actividades para el buscador de contenidos.

Los diagramas de las Figuras 3.2, 3.2 y 3.3 son fragmentos de los originales y que están cambiadas palabras que comprometan la confidencialidad del sistema, ya que no es posible describirlos o mostrarlos completos por violaciones de lo establecido, sin embargo estos detallan de manera general los requerimientos solicitados por la empresa que son: poder insertar cadenas de texto, uso de filtros para búsquedas especializadas y regresar el resultado esperado. Este análisis aportó las siguientes características: se encontraron las relaciones entre los componentes y actores del sistema, las actividades que realiza cada actor y los procedimientos que siguen para lograr la búsqueda. Estos resultados son de vital importancia para el desarrollo del buscador de contenidos educativos y para este reporte.



3.3.2 ANÁLISIS DE LA CONFIGURACIÓN

En este análisis se revisó la infraestructura del sistema educativo, con la intención de verificar que éste debería poder ser ejecutado en dispositivos móviles (*tablets*) y computadoras personales a través de internet. Las características de las *tabletas* utilizadas se muestran en la Tabla 3.4.

Tabla 3.4 Dispositivos móviles utilizados para el desarrollo del buscador, estos dispositivos fueron adquiridos para la etapa de pruebas.

| Nombre | Sistema Operativo | Características |
|--------------------|-------------------|-----------------------------------------|
| Samsung Galaxy Tab | Android 4.2 | Wifi Pantalla 10.1” Cámara 3 MP |
| iPad | IOS | Wifi Pantalla 9.7” Cámara no |
| iPad mini | IOS | Wifi Pantalla 7.9” Cámara 5 MP HD |
| Surface Pro | Windows 8.1 | Wifi Pantalla 12” Cámara 5 MP |

Derivado de este análisis se determinaron la base de datos a utilizar y servidor web, mismos que se analizaron en el capítulo dos, los cuales son PostgreSQL y Apache.

3.4 INGENIERÍA PARA EL DESARROLLO DEL BUSCADOR

Esta etapa consistió en la elaboración del diseño del sistema, el cual resuelve el problema que se planteó; se definió una arquitectura que tenga la facilidad de evolucionar con el tiempo. Uno de los principales principios de diseño son el uso de heurísticas, y una de estas es la modularidad, esto se llevó a cabo para la reutilización de módulos. Otro factor son las configuración del diseño para resolver problemas específicos que se presentaron durante



esta etapa y el uso de plantillas fue importante para este módulo de búsqueda, debido a que se incorporó en todas las interfaces del sistema educativo.

Para esta etapa la metodología iWeb (ver anexo A) propone realizar cinco diseños los dos primeros son el diseño del contenido y producción los cuales se realizan en paralelo con los diseños arquitectónico, navegación y de la interfaz, los cuales se describen en las siguientes secciones.

3.4.1 DISEÑO ARQUITECTÓNICO

Para el diseño arquitectónico se definió la estructura global de toda la aplicación, se definió también los usuarios a los que está dirigido el módulo de búsqueda los cuales son alumnos de nivel primaria y preescolar, estos usuarios pueden en cualquier momento ejecutar contenidos educativos sin necesidad de que estén inscritos en algún curso.

Durante esta etapa se debe realizar la configuración del diseño, donde se llevaron a cabo el control de versiones, tanto para las versiones del sistema como para las versiones de la base de datos, las cuales ayudaron a resolver problemas que se presentaron durante el desarrollo. Uno de estos problemas fue que al desplegar una versión del módulo de búsqueda esta podría fallar, de tal manera que el control de versiones ayudó a determinar la versión del módulo que estaba dañada o funcionando mal.

En este diseño se determina la estructura del sistema que fue una estructura jerárquica, en la que casi todas las páginas pueden hacer referencia a la página del buscador, ya que los resultados se dirigen a ésta (ver Figura 3.4). Con este diseño se facilitó la modificación de la interfaz y la reutilización de código sin modificar las páginas ya existentes.

Como se puede apreciar en la Figura 3.4, otros módulos y páginas de la GUI del usuario pueden ingresar al buscador, pero también páginas que usen las plantillas hechas en PHP para el buscador. De tal manera que el uso de plantillas en este diseño fue fundamental, ya que este módulo debería estar en todos los demás módulos y éstas se realizaron en PHP para que fueran dinámicas.

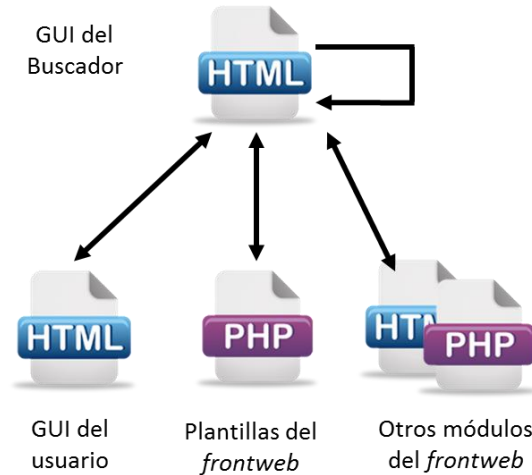


Figura 3.4 Estructura jerárquica y navegación, utilizada para el desarrollo del buscador de contenidos [9].

3.4.2 DISEÑO DE LA NAVEGACIÓN

Una vez establecida la arquitectura, se diseñó la ruta de la navegación. La ruta de navegación que se estableció para el buscador fue simple ya que solo hay una dirección, todas las páginas de la GUI de usuario, Plantillas y otros módulos del *frontweb* pueden dirigirse al buscador en cualquier momento, como se muestra en la Figura 3.4. Además, una vez estando en la interfaz del buscador solo se puede navegar hacia las interfaces principales, y también cuando se realice otra consulta se quedará en la misma interfaz del buscador.

Dado que la estructura es jerárquica la navegación puede ser hacia dos sentidos, siempre y cuando las páginas que estén relacionadas tengan un vínculo o evento hacia la otra página, en la GUI del buscador las interfaces que pueden acceder a él y viceversa son las interfaces principales (menús).

La navegación por lo regular es lo primero que se diseña y una vez que se tiene se realiza la estructura del sistema, estas malas prácticas no se deben realizar primero es la estructura y después la navegación, dado que se siguió la metodología de una manera estricta primero se diseñó la estructura, que solo contiene flechas a las interfaces que puede acceder a través de vínculos, como en este caso las únicas interfaces que no acceden al buscador son las de inicio y cierre de sesión, todas las demás del *frontweb* si tiene acceso.



Por lo tanto la navegación detalla la estructura que se tiene y que por motivos de confidencialidad se mostró de manera muy general en la Figura 3.4.

3.4.3 DISEÑO DE LA INTERFAZ

Para diseñar la interfaz del buscador de contenidos educativos se basó en la interfaz de Google, quien cuenta con una interfaz simple, permitiendo enfocarse en lo que se quiere buscar, gracias a que solamente contiene una caja de texto acompañada de un botón con lo que se inicia la búsqueda, también este cuenta con categorías de búsqueda a través de un menú emergente, que son Web, Imágenes, Mapas, Videos, Noticias, Libros y Aplicaciones. Cada uno de estos temas tiene sus filtros de búsqueda avanzada [12].

La interfaz se diseñó en base a los requerimientos descritos en la Tabla 3.3 pero también con los puntos de vista de los usuarios, en el ejemplo anterior y en desarrolladores del sistema, que estuvieron involucrados durante el ciclo de vida del mismo. En el diseño inicial los campos eran estáticos y siempre era los mismos, este fue uno de los principales errores de diseño ya que el usuario podría elegir opciones de búsqueda sin relación y que regresarán resultados nulos o incongruentes.

El diseño propuesto para la interfaz del buscador de contenidos fue el que se muestra en la Figura 3.5, el cual consta con un apartado para la inserción del texto y dos botones. El primero para las opciones de los filtros y el segundo para la búsqueda. Al diseño de esta interfaz únicamente se le añadió el icono de filtros de búsqueda.

La interfaz para la parte de filtros de búsqueda tuvo dos variantes la primera que se basó en el despliegue de un menú emergente (ver Figura 3.5.a), pese a que este diseño resultó novedoso, se encontraron problemas con la compatibilidad en las librerías de las demás páginas y que en dispositivos móviles no se respetaba, de tal manera que entre mayor fuera la resolución de la pantalla, esta interfaz no se adaptaba, mandando elementos de la GUI fuera de la pantalla.

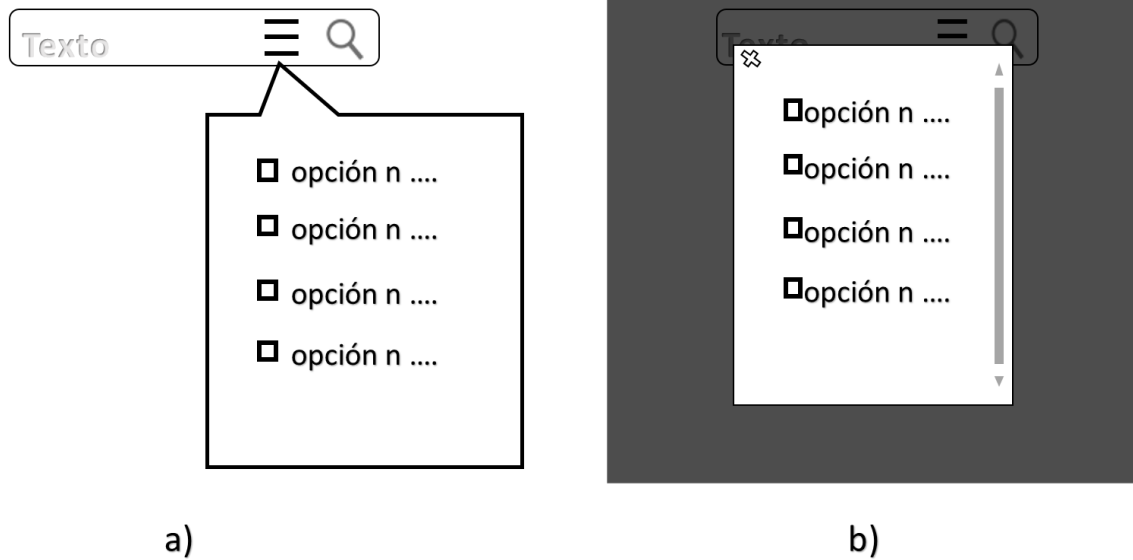


Figura 3.5 Diseños de prototipos para la interfaz del buscador, a) Diseño de la interfaz utilizando un menú emergente. b) Diseño de la interfaz utilizando una pantalla modal.

Para evitar el problema de compatibilidad con las librerías de JavaScript se diseñó la interfaz mostrada en la Figura 3.5.b, la cual consistía en el llamado de una pantalla modal al hacer clic en el botón de filtro. La pantalla modal es simplemente mostrar todo la interfaz del usuario inactiva con un color más oscuro y únicamente como una pantalla activa el selector de filtros como se muestra en la Figura 3.5.b. Esta última interfaz fue compatible con la mayoría de los dispositivos móviles requeridos por el usuario. Otra adecuación fue la colocación de un panel deslizable con un conjunto de botones con opciones asignadas de forma dinámica utilizando funciones en JavaScript que precargan las opciones dependientes entre sí, cuando el usuario hacia clic sobre ellas,

Las diferencia fundamental entre estas dos interfaces es que en la del menú emergente el usuario podría interactuar con lo demás ya sea íconos, botones que no fueran parte de la interfaz del buscador, y en la segunda interfaz que es la de la pantalla modal el usuario solo podía interactuar con la interfaz de los filtros, evitando así que se distrajera con otro componente de GUI.



3.4.4 DISEÑO DEL CONTENIDO Y PRODUCCIÓN

El diseño de contenido es una actividad no técnica por lo que fue desarrollada por los diseñadores gráficos de la empresa, para la generación de diseños de imágenes, videos, tipografías, multimedia, etc., que generaron los contenidos del sistema educativo. De esta forma el contenido presentado en el sistema educativo tiene un carácter profesional ya que son realizados por expertos en el área.

En el diseño de la producción se generan lo anterior de la etapa de diseño del contenido, que es en donde entran los diseñadores gráficos de la empresa, creando los contenidos educativos los cuales deben de seguir un estándar, que consiste en el uso de funciones JavaScript definidas y llamadas en el código por los módulos desarrollados por la FI. Al final de la etapa los contenidos que se suben son dinámicos y pueden tener opciones más complejas como la obtención de estímulos o logros por parte de los usuarios.

Estas etapas se llevaron en paralelo con las demás etapas de Ingeniería, de manera que los contenidos que se diseñaron en esta etapa podrían ser validados desde todos los perfiles, esto llevo una ventaja para prevenir errores en la ejecución de los contenidos educativos. Ya que probar las funciones de los contenidos y que estos se ejecutarán correctamente facilitó el desarrollo del buscador y de otros módulos.

3.5 GENERACIÓN DE PÁGINAS

Es la etapa de construcción del diseño y funcionalidad descrita en la etapa de ingeniería. Para esto se programa la funcionalidad de las interfaces del buscador, implementando los diseños descritos anteriormente. Se utilizó una programación basada en el patrón arquitectónico de tres capas, dividiendo el módulo de búsqueda en las capas de presentación (vista al usuario), la capa de negocios (lógica) y la capa de datos (base de datos) [23]. De esta forma cada página se programa en la parte del *frontweb*, que es donde está toda la GUI de usuario y las funciones en un controlador en el que todas las funciones son llamadas y definidas (ver Figura 3.6).

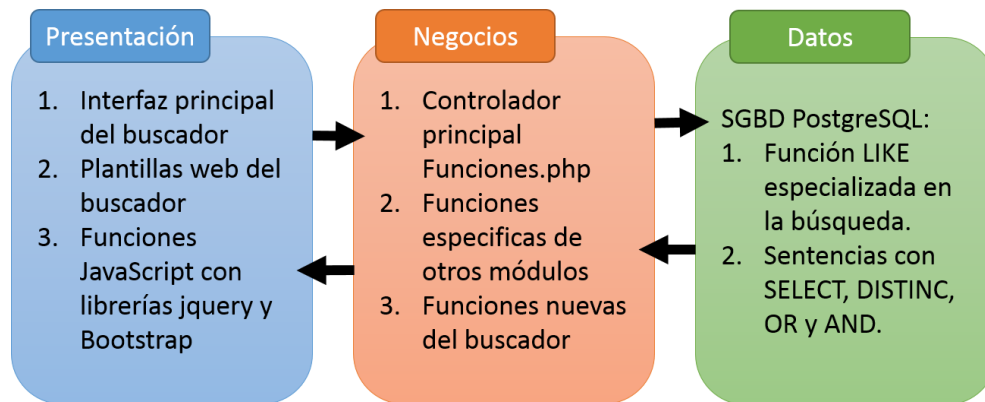


Figura 3.6 Componentes utilizados para el patrón arquitectónico de tres capas [44].

Algunos detalles de los componentes que se utilizaron para el desarrollo del buscador siguiendo el patrón arquitectónico de tres capas:

1. **Presentación.** Para esta capa se programó la interfaz principal del buscador en PHP con HTML en su versión cinco y JavaScript con las librerías de jQuery y Bootstrap, para crear la interfaz de la pantalla modal ilustrada en la Figura 3.5.b. Esta interfaz contenía plantillas en PHP, estas plantillas servían únicamente para cambiar el color de la interfaz según el nivel escolar, y para que las demás interfaces de los otros módulos pudieran introducir la interfaz del buscador. Esta interfaz está contenida en la parte del *frontweb*, en donde estaban tanto las imágenes e iconos, así como los contenidos multimedia que usaba página del buscador.
2. **Negocios.** En esta capa se programaron las funciones de la lógica de la aplicación para el buscador, haciendo uso de consultas de la base de datos se utilizando SQL con las funciones del controlador, las cuales perfeccionaron la búsqueda regresando el resultado a la parte de presentación en forma de lista para que el usuario ejecute los contenidos. Todas estas funciones deberían seguir el patrón de codificación *module*, en donde se tenía una función principal llamada “Funciones.php” que mandaba a llamar a las otras funciones y también en donde se definían las nuevas funciones, este patrón garantizó que otros módulos pudieran reutilizar las funciones creadas para el buscador.
3. **Datos:** En esta capa la base datos PostgreSQL utiliza la sentencias SELECT, DISTINCT, UNION, LIKE para crear una búsqueda específica, en especial la



sentencia LIKE que busca palabras que estuvieran contenidas en campos de la base de datos como si fueran lenguaje natural.

El uso de estos patrones de software evito que la programación se realizara de una manera no uniforme, es decir que no tuviera un orden y que no se visualizara la relación que hay entre los componentes de la aplicación, como la que se muestra en la Figura 3.6, en donde el patrón arquitectónico de tres capas dicta una solución muy organizada para programar el buscador de contenidos educativos. Así como el patrón de codificación module, utilizando en la capa de Negocios para llevar una estructura de las funciones de la aplicación.

Un factor importante durante la generación de las páginas, fue que las funciones utilizadas en la capa de negocios que consultaban a la base de datos estaban limitadas, ya que PostgreSQL no cuenta con funciones como las que tiene otros manejadores de BD que mejoran la búsqueda, sin embargo cuenta con expresiones regulares que en posteriores desarrollos podrían crear una búsqueda más especializada.



CAPÍTULO 4. EVALUACIÓN

En este capítulo se describirán las últimas dos etapas de la metodología iWeb que se utilizaron para la parte de evaluación. En donde se realizaron correcciones en cada iteración del sistema a los resultados arrojados por las pruebas. De esta manera se minimizaron errores, los cuales producirían costos de esfuerzo y desarrollo.

4.1 PRUEBAS

La metodología propone realizar varias pruebas al sistema que se describen más adelante, todas éstas llevadas a cabo por los evaluadores del equipo de desarrollo de la FI, las cuales consistieron en minimizar errores en el código, en la estética del sistema y en la parte lógica, este tipo de pruebas por lo general son llamadas por muchos desarrollados como pruebas de caja blanca y caja negra [33]. Las pruebas de caja blanca permiten encontrar errores lógicos en el código, así como bugs, y las pruebas de caja negra son más enfocadas a errores de estética relacionados con la interfaz de usuario [9]. Sin embargo las pruebas que utiliza la metodología iWeb son más específicas y de detallan a continuación.

1. Pruebas hacia el modelo del contenido. En estas pruebas es donde se descubrieron errores gramaticales, tipográficos, errores de consistencia en el buscador, errores de las representaciones gráficas como el menú de los filtros y mensajes mostrados en las páginas ya sea de error o de información.
2. Pruebas al modelo del diseño. En estas pruebas se evalúa el diseño arquitectónico y la navegación del buscador donde se encontraron errores principalmente en la parte de la navegación.
3. Las pruebas de unidad. Se realizaron a las páginas web por separado, como unidad mínima, esta tarea es algo laboriosa ya que como el buscador está contenido en la mayoría de las páginas del *frontweb*.
4. Las pruebas de integración. Estas tienen que ver con la estructura que se definió en el diseño arquitectónico. Como la estructura es jerárquica es bastante simple, todas las páginas del *frontweb* tiene acceso al buscador, de esta manera los errores encontrados en esta prueba fueron escasos o nulos.



5. Las pruebas de validación. Se realizaron de manera detallada, para reconocer escenarios posibles de error de requisitos de interacción con el usuario, para esto el equipo de la FI participó detectando errores y posteriormente reportándolos a los evaluadores.
6. Las pruebas de entornos y compatibilidades. Estas pruebas fueron realizadas, probando el buscador de contenidos en diferentes dispositivos, en este caso se utilizaron tabletas con diferente sistema operativo, en los cuales se detectó incompatibilidad con algunos navegadores y que se corrigieron cambiando unas librerías en el código como se muestra en la Tabla 4.1.
7. Las pruebas con muestra de usuarios. Se realizaron para determinar errores de contenido y navegación en el buscador relacionados con la usabilidad, compatibilidad, fiabilidad y rendimiento del buscador de contenidos con los usuarios.

Todas estas pruebas se realizaron en conjunto con los evaluadores del equipo de desarrollo de la FI, llevando a cabo la detección de errores. Los resultados de las pruebas antes mencionadas se describen en la Tabla 4.1. El tiempo que se llevó para las pruebas fue corto y que se describen en el plan de trabajo, mismos que no pueden ser descritos por motivos de confidencialidad.

Estas pruebas fueron más detalladas por los evaluadores ya que ese era su perfil, además de documentar las prueba utilizando la metodología MAGTIC de la empresa, la Tabla 4.1 muestra de manera general algunas pruebas que se llevaron a cabo, en donde se utilizaron cadenas como “mate” ya que muchos contenidos contenían ese fragmento de su nombre o descripción. Estas pruebas son únicamente las pruebas en la que se participó y que fueron de vital importancia para la aplicación de la metodología iWeb.

Otras pruebas que se realizaron fueron en el momento de la programación del módulo de búsqueda, en las cuales los errores que se encontraron eran lógicos, como el uso incorrecto de las sentencias a la base de datos, el mal uso de nombre de las variables así como errores de sintaxis y ortográficos que eran visibles para el programador.



Tabla 4.1 Pruebas realizadas el buscador en diferentes dispositivos móviles y ambientes.

| Dispositivo | Prueba | Detalle | Resultados |
|--------------------|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| iPad , iPad Mini | Entornos y compatibilidades | Realización de una búsqueda con la palabra “mate”, se ejecutó el primer contenido encontrado. | Los resultados fueron los esperados, además el contenido se ejecutó en pantalla completa. |
| Samsung Galaxy Tab | Entornos y compatibilidades | Realización de una búsqueda con la palabra “mate”, se ejecutaron todos los contenidos encontrados. | Los resultados no fueron los esperados. Los contenidos no se podían visualizar por el usuario. |
| Laptop y PC | Integración | Ejecución del buscador desde todas las interfaces de otros módulos que lo contenían. | Los resultados fueron los esperados. De acuerdo a la estructura del sistema, solo usuarios con perfil de alumno podrían entrar al sistema, ningún otro perfil tenía acceso. |
| Laptop y PC | Modelo del Diseño | Se comprobó toda la navegación, ejecutando vínculos a otras páginas desde el buscador, y desde los otros módulos hacia el buscador. | En los resultados se encontró un error cuando se estaba en la misma página del buscador. No esta validado y mostraba un mensaje no entendible para el usuario. Es decir era el resultado de una sentencia. |
| Todos | Unidad | En cada interfaz o módulo que contenía el buscador se realizó esta prueba, utilizando cualquier palabra para determinar su funcionamiento y la ejecución del mismo. | Algunos módulos utilizaban el mismo nombre de etiquetas, lo cual hizo que alguno de los dos módulos dejara de funcionar, o bien que tenían librerías no compatibles con las del buscador, utilizaban una versión mayor o menor a las librerías de jQuery. |



4.2 EVALUACIÓN DEL CLIENTE

Es la última etapa de la metodología se llevaron a cabo todas las correcciones y cambios que los evaluadores asignaron a los programadores, de los errores encontrados y que se dieron solución fueron los que se describen en la Tabla 4.2. Esta etapa fue fundamental para que el sistema fuera el esperado por el cliente, de manera que la etapa de pruebas de la metodología iWeb mostro los errores y que se debería cambiar.

Tabla 4.2 Solución a los errores y cambios encontrados en la etapa de pruebas.

| Lugar | Error | Solución |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Interfaz principal | Errores en dispositivos móviles, en los cuales la interfaz se distorsionaba mandando los elementos del buscador fuera del área de trabajo. | Utilizar librerías JavaScript compatibles (jQuery) con la interfaz de usuario. Usar estilos CSS dinámicos para cada resolución de los dispositivos móviles. |
| Base de datos | Los resultados arrojados no coincidían con el patrón de búsqueda, debido a que se hacía un mal uso de la sentencia LIKE de PostgreSQL, esta se combinaba con la sentencia UNION la cual regresaba la mezcla de consultas. | Se cambió la sentencia UNION por los operadores lógicos AND y OR para crear una única búsqueda con relaciones entre los demás campos en la base de datos PostgreSQL. |
| Interfaz de Filtros | La interfaz de filtros permitía elegir opciones que no tenían relación alguna mostrando resultados inconsistentes. | Se utilizaron funciones JavaScript con jQuery para precargar opciones de los filtros únicamente relacionados, con una usabilidad sencilla. |
| Nombre de las Variables | El nombre de las etiquetas en HTML y de las variables en JavaScript chocaba con el nombre de otras variables de otro módulo que hacía uso de las mismas funciones. | Se cambió el nombre de las variables, para que le módulo de búsqueda tuviera mayor prioridad. |
| Base de datos | Se solicitó cambiar la estructura de la tabla que guarda los contenidos en la base de datos PostgreSQL para añadir una nueva funcionalidad. | El cambio realizado afecto varias funciones por lo cual estas se modificaron para que se adaptaran a la nueva tabla. |



En cada iteración de la metodología iWeb se encontraron diferentes errores y las correcciones de la Tabla 4.2 son de manera general todas las que surgieron durante este desarrollo, de manera que para cada incremento los errores encontrados fueron mínimos, pero también la empresa añadió más requerimientos los que gracias al uso de esta metodología se pudo dar solución a los cambios y al módulo de búsqueda de contenidos educativos para su ejecución individual.



CONCLUSIONES Y SUGERENCIAS

Partiendo del objetivo planteado, al finalizar las actividades encomendadas en el proyecto, fue posible la entrega del buscador de contenidos funcionando de forma consistente con apego a los requerimientos establecidos por la empresa. De tal modo que el buscador, permite la ejecución de contenidos educativos de manera independiente y sin restricciones, con funcionamiento en web y en dispositivos móviles.

Respecto al desarrollo de las diferentes etapas del buscador con apego a la metodología iWeb se observó que es fundamental contar con un seguimiento puntual de ésta para hacer de manera ordenada el desarrollo de sistemas. Permitiendo conocer a detalle lo que el usuario necesita, así como el conocimiento de otros sistemas que se tengan ya hechos, en muchas ocasiones la reutilización de módulos es útil para los programadores y los desarrolladores del sistema.

Las herramientas de desarrollo que se utilizaron y analizaron en el capítulo dos fueron las más adecuadas, debido a la compatibilidad que se tenía con otros módulos, y también porque se siguió este mismo esquema de programación. Sin embargo se presentaron algunos inconvenientes en la integración con el código desarrollado por ITT, como son el uso de variables que no eran necesarias y la repetición de fragmentos de código con funcionalidad similar.

Algunas buenas prácticas rescatables son la importancia de un equipo de trabajo integrado y bien informado para resolver un problema real, así como una buena comunicación ya que para este desarrollo cada becario realizó diferentes módulos y algunos de estos dependían de otros. Además en la etapa de pruebas se fortalecieron mis conocimientos y habilidades como ingeniero para detectar errores y resolverlos.

Algunas sugerencias que puedo verter son que el buscador desarrollado puede ser utilizado para regresar resultados personalizados como podrían ser libros, revistas, cursos, imágenes, e incluso el mismo sistema puede crear una página web en función del resultado, de esta manera se pueden realizar búsqueda más dinámicas que regresan contenidos



como multimedia, videos, mapas, juegos, etc., los cuales se construyen a partir del resultado de la búsqueda para futuros desarrollos.

Finalmente, considero que este desarrollo fue fundamental para mi formación como Ingeniero en Computación gracias a que se llevó un estricto seguimiento de la metodología iWeb. Debido al conocimiento aplicado durante este tiempo, que fueron el análisis y modelado de sistemas que ayudaron a comprender los requerimientos del buscador, el diseño y desarrollo de sistemas en la cual se pusieron a prueba las habilidades como programador para entregar este producto y por último en la parte de pruebas y correcciones fue extra para mi formación, ya que las pruebas no son mi fuerte, y que al final son indispensables para el desarrollo de sistemas.



REFERENCIAS

- [1] Lamarca Lapuente María Jesús. (2013). *Hipertexto: el nuevo concepto de documento en la cultura de la imagen*. Tesis doctoral, Departamento de Biblioteconomía y Documentación, Universidad Complutense de Madrid.
- [2] Torres Pombert Ania. (2003). El uso de los buscadores en Internet. *ACIMED*, 11, 3-8.
- [3] Francisco Javier Martínez Méndez. (2002). *Propuesta y desarrollo de un modelo para la evaluación de la recuperación de información en internet*. Tesis doctoral, Departamento de Información y Documentación, Universidad de Murcia.
- [4] Castrillón Estrada Jaime A., García Domínguez Juan Camilo, Anaya Taboada Marco, Rodríguez Berdugo Deisy, de la Rosa Barranco Dereck, Caballero-Uribe Carlo V. (2008). Bases de datos, motores de búsqueda e índices temáticos: herramientas fundamentales para el ejercicio médico. *Revisiones Clínicas*, 24, 96-119.
- [5] Martínez Ángeles Maldonado y Rodríguez Yunta Luis. (2006). *La Información especializada en Internet: directorio de recursos de interés académico y profesional*. Madrid: CSIC.
- [6] Stark Natalia S. (2001). *Motores de búsqueda en internet*. Teleinformática y Redes. Universidad Nacional de Luján.
- [7] López de Armentia S. Lapeña, Álvarez Ramos R., Ledesma Benítez I. (2006). Motores de búsqueda y bases de datos médicas. *BOL PEDIATR*, 46,217-220.
- [8] Borrero Caldas María Clara, Cruz García Edgar, Mayorga Muriel Sandra, Ramírez González Karen. (2008). *Una metodología para el diseño de objetos de aprendizaje*. Dirección de Nuevas Tecnologías y Educación Virtual. Universidad del Valle.
- [9] Pressman Roger. (2010). *Software engineering. A Practitioner's Approach*. Madrid: McGraw-Hill.
- [10] Heurtel Oliver. (2011). *PHP 5.3 Desarrollar in sitio web dinámico e interactivo*. Barcelona: Ediciones ENI.
- [11] Aguillo Isidrio F. (2002). Del multibuscador al metabuscador: Los agentes trazadores de internet. *CINDOC-CSIC*, 22, 3-10.
- [12] Suarez Sánchez Alejandro. (2012). *Desnudando a Google*. España: Deusto.
- [13] Torres E. (2001). *¿Qué son las bases de datos? Herramientas para la investigación*. Barranquilla: Ediciones Uninorte.



- [14] Quintana G., Marques M., Aliaga J. L., Aramburu M. J. (2008). *Aprende SQL*. España: Uniservitat Jaume.
- [15] Fernández Baizán Covadonga. (2004). *El modelo relacional de datos: de los fundamentos a los modelos deductivos*. Madrid: Ediciones Díaz de Santos, S. A.
- [16] Viñao Antonio. (2006). *Sistemas educativos, culturas escolares y reformas*. Madrid: Ediciones Morata, S. L.
- [17] Francisco García García. (2006). Contenidos educativos digitales: Construyendo la Sociedad del Conocimiento. *Red Digital*, 6, 1-29.
- [18] Mohammed J. Kabir. (2002). *La biblia del servidor apache 2*. Madrid: Editoriales Anaya.
- [19] Gutiérrez Emmanuel. (2009). *JavaScript conceptos básicos y avanzados*. Barcelona: Edit. ENI.
- [20] Gauchat Juan Diego. (2012). *El gran libro de HTML, CSS3 y JavaScript*. Barcelona: Edit. Marcombo.
- [21] Nevado Cabello Victoria. (2008). *Introducción a las bases de datos relacionales*. Madrid: Vision libros.
- [22] PostgreSQL. *Sobre PostgreSQL* (en línea). Obtenido de: http://www.postgresql.org.es/sobre_postgresql, consultado: abril 2015.
- [23] García González Roberto. Patrones Arquitectónicos de Aplicaciones Empresariales. *Universidad de Lleida*, 1, 2-5.
- [24] Fowler Martin y Rice David. (2003). *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley.
- [25] Damián Campo Gustavo. (2009). Patrones de Diseño, Refactorización y Antipatrones. Ventajas y Desventajas de su Utilización en el Software Orientado a Objetos. *Cuadernos de la Facultad*, 4, 103-105.
- [26] Lorena Lazo Jane Paul. (2004). *Desarrollo de sistema de software con patrones de diseño orientados a objetos*. Tesis de licenciatura, Universidad Nacional Mayor de San Marcos, Lima, Perú.
- [27] Buschmann Frank, Meunier Regine, Rohnert Hans, Sommerlad Peter y Stal Michael. (1996). *Pattern-Oriented Software Architecture - A System Of Patterns*. U.S.: John Wiley & Sons.
- [28] Guaniapa M, (2009). *Sistema bajo Tecnología web para el control de asistencia del personal Adscrito a la unidad de apoyo punto fijo de la Universidad Nacional Abierta*



- (UNA). Tesis de Licenciatura, Universidad Nacional Abierta (UNA), Caracas, Venezuela.
- [29] Matéu C (2004). *Software Libre: Desarrollo de Aplicaciones web*. Tesis de Maestría, Universidad Oberta de Catalunya, Catalunya, España.
- [30] Quiroz, I. (2007). *Desarrollo de aplicaciones para la construcción de sitios interactivos en Internet para el comercio electrónico*. Tesis Licenciatura, Departamento de Ingeniería en Sistemas Computacionales, Escuela de Ingeniería, Universidad de las Américas Puebla.
- [31] Pérez Iglesias Mónica. (2012). *Google: Como construir la mayor base de datos del mundo*. España: Editorial PROFIT
- [32] Dolores Alcaín María. (2000). *Recursos para Farmacéuticos en internet: Manual de Fuentes de información*. Madrid: CSIC
- [33] Fitzgerald Michael y Pasha Ali. (2007). *Google Ajax Search API. O'REILLY Short Cuts*. 12, 8-18.
- [34] Fontán Baquero Juan. (2013). *Sistemas de información y bases de datos en consumo*. Pontevedra: Certia Editorial
- [35] Vilarroya Óscar. (2006). *Palabra de Robot, Inteligencia artificial y comunicación*. Valencia: Maite Simón
- [36] Siri Laura. (2000). *Internet: búsquedas y buscadores*. Buenos Aires: Editorial Norma.
- [37] Calvo Verdú Miguel. (2005). *Formación Profesional Ocupacional*. España: Editorial Mad.
- [38] Universidad Autónoma del Estado de México. *Repositorio Institucional*. (En línea). Obtenido de: <http://ri.uaemex.mx/>, consultado: mayo 2015.
- [39] Heurtel Oliver. (2014). *PHP 5.5: Desarrollar un sitio Web dinámico e interactivo*. Barcelona: Editorial ENI.
- [40] Lavinge Dru. (2004). *BSD Hacks: 100 Industrial Tips and Tools*. E.U.: O'REILLY.
- [41] Worsley, John C, Drake, Joshua D. (2002). *Practical PostgreSQL*. E.U.: O'REILLY.
- [42] Netcraft. *May 2015 Web Server Survey*. (En línea). Obtenido de: <http://news.netcraft.com/>, consultado: mayo 2015.
- [43] Marcia Perez Francisco. (2008). *Administración de servicios de internet: De la teoría a la práctica*. Mallorca: Compobell
- [44] Freeman Eric y Freeman Elisabeth. (2004). *Head First Desing Patterns*. E.U: O'REILLY.



- [45] Lerdorf Rasmus, Tatroe Kevin y MacIntyre. (2014). *Programming PHP*. E.U.: O'REILLY.
- [46] Cortes Marc. (2009). *Nanoblogging: Los usos de las nuevas plataformas de comunicación en la red*. Barcelona: Editorial UOC.
- [47] Taverniti Giorgio. (2012). *SEO power. Strategie e strumenti per essere visibili online*. Italia: Hoepli
- [48] Cavaller Víctor, Pedraza Rafael, Codina Lluís, Sánchez Silvia. (2014). *Estrategias y gestión de la comunicación online y offline*. Barcelona: Editorial UOC
- [49] Pérez Cesar. (2014). *Técnicas de minería de datos e inteligencia de negocios*. Barcelona: Editorial Garceta
- [50] Vieira Braga Luis Paulo, Ortiz Valencia Luis Iván y Segundo Santiago. (2009). *Introducción a la Minería de Datos*. Brasil: E-papers
- [51] Geschwinde Ewald y Schönig Hans-Jürgen. (2002). *PostgreSQL Developer's Handbook*. E.U.: SAMS

ANEXO A - METODOLOGÍA IWEB

Para el desarrollo del proyecto en su conjunto, el equipo de desarrollo de la Facultad de Ingeniería propuso usar la metodología iWeb [9]. Esta metodología consta de seis etapas (ver Figura A.1) que manejan un proceso incremental y evolutivo, lo que la convierte en un modelo eficiente para el desarrollo de sistemas web. Las siguientes secciones brindan una descripción detallada de cada etapa.

1. **Formulación.** En la etapa de Formulación se identifican las metas y los objetivos del sistema, estableciendo de este modo la motivación del desarrollo del sistema, su importancia y los usuarios potenciales.
2. **Planificación.** En la etapa de planificación, se estima el costo global del proyecto y se evalúan los riesgos asociados con el esfuerzo del desarrollo, y se define una planificación del desarrollo muy detallada para el incremento final de la aplicación. De esta manera la planificación para los incrementos siguientes es más específica.
3. **Análisis.** En esta etapa se establecen los requisitos técnicos y de diseño, e identificación de los elementos de contenido que se van a incorporar. Durante esta etapa se realizan cuatro tipos de análisis diferentes [9].
 - **Análisis del contenido:** Se identifica el aspecto completo del contenido que se va a proporcionar, este contenido incluye datos de texto, gráficos, imágenes, videos y sonido, utilizando un modelado de datos.
 - **Análisis de la interacción:** Se trata de la descripción detallada de la interacción del usuario, a través de casos de uso prácticos.
 - **Análisis funcional:** Los casos de uso descritos en el análisis anterior, definen operaciones y funciones que se aplican al contenido del sistema, las cuales se detallan.
 - **Análisis de la configuración:** Se realiza una descripción detallada del entorno y de la infraestructura del sistema.

4. Ingeniería. En esta etapa se realizan las tareas diseño del contenido y producción, en paralelo con los diseños arquitectónicos, navegación e interfaz [9].
 - Diseño arquitectónico: Este diseño se realiza en paralelo con el diseño del contenido, en los cuales se centra en el diseño de la estructura global del sistema, así como en las configuraciones del diseño y plantillas.
 - Diseño de navegación: Se identifica la semántica y la sintaxis de la navegación, identificando los diferentes perfiles que se establecieron y que navegación tiene cada uno de ellos.
 - Diseño de la interfaz: En este diseño se realizan todos los ajustes para que la interfaz de usuario sea la ideal, evitando factores como que el usuario abandone el sitio web, el tamaño del texto, etc.
 - Diseño del contenido y de la producción: Son tareas que se llevan a cabo por personas no técnicas, el propósito de éste, es el de diseñar o adquirir todo el contenido de texto, gráfico, imágenes y video que se van a utilizar en el sistema.

5. Generación de páginas. En esta etapa se realiza la construcción haciendo uso de las herramientas para el desarrollo de aplicaciones web, sistemas y se asocia con el diseño arquitectónico, de navegación y de interfaz para la elaboración de web dinámicas.

6. Pruebas. En esta etapa se busca descubrir errores y ayuda a asegurar que la aplicación web funcionará correctamente en diferentes entornos. Para esto se hace uso de estrategias y técnicas que hayan sido recomendadas para otros sistemas [9].
 - El modelo del contenido, es una prueba que se realiza para detectar errores ortográficos.
 - El modelo del diseño, es revisado para descubrir errores en la navegación, en este caso se proponen escenarios para descubrir lo posibles errores.
 - Las pruebas de unidad se realizan a cada página para encontrar errores más específicos.

- Las pruebas de integración, evalúan la estructura que se definió en la arquitectura que se haya elegido para el sistema.
 - Unas pruebas comunes son las de validación, las cuales se basan en casos prácticos proporcionando escenarios con una probabilidad alta de cubrir todos los errores.
 - En las pruebas de compatibilidad y configuración, se definen todas las posibles plataformas de hardware para los navegadores donde se visualizará el sistema y los protocolos de comunicación.
 - Las pruebas de control y monitorización se aplican a todos los usuarios posibles del sistema y se evalúan los resultados de su interacción con el sistema.
7. Evaluación del cliente. En esta etapa es donde se realizan todas las correcciones y cambios que se detectaron en la etapa de pruebas y se integran al sistema para el siguiente incremento, de tal modo que se asegure la satisfacción por parte del cliente, según los requerimientos solicitados.

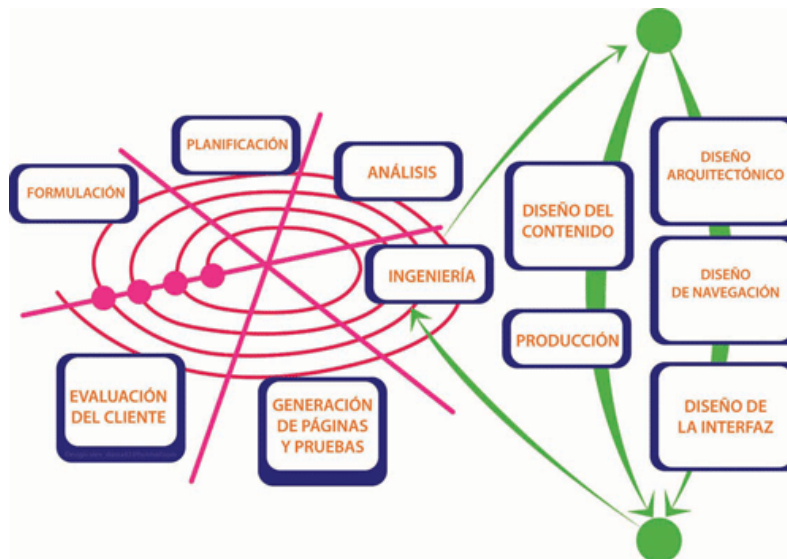


Figura A.1 Diagrama del ciclo de vida de la metodología iWeb [9].

Adicional a la evaluación en esta etapa se realiza la transferencia tecnológica del sistema desarrollado, es decir, se realiza el alojamiento en servidores o en los equipos que para ello el cliente considere pertinente.

ANEXO B – CARTA DE ACEPTACIÓN



México D.F. a 29 de mayo de 2015.

A QUIEN CORRESPONDA:

Por medio de la presente se hace constar que la empresa **e-Blue S.A. de C.V.** deslinda de toda responsabilidad a **Víctor Hugo Solís Ramos**, así como a la **Facultad de Ingeniería** de la **Universidad Autónoma del Estado de México (UAEMex)**, de mostrar información confidencial en su trabajo de titulación **“Aplicación de la Metodología iWeb en el desarrollo de un módulo web para la búsqueda de contenidos en un sistema educativo”**, generado como producto de investigación de su participación como becario en el proyecto **Espacio Digital para el Aprendizaje Autónomo: MetaSpace**.

Habiendo previamente revisado el trabajo de titulación y aceptado que el contenido no compromete información confidencial, se extiende este documento a los 29 días del mes de mayo del 2015.

Saludos cordiales.

Ing. Horacio Alejandro Camacho Martínez

Líder del Proyecto



CARTA DE CESIÓN DE DERECHOS DE AUTOR

El que suscribe Victor Hugo Solis Ramos Autor(es) del trabajo escrito de evaluación profesional en la opción de Rep. Apl. Canam con el título Aplicación de la Metodología Web en el desarrollo de un módulo por medio de la ~~para la búsqueda de contenidos en un sistema educativo~~ presente con fundamento en lo dispuesto en los artículos 5, 18, 24, 25, 27, 30, 32 y 148 de la Ley Federal de Derechos de Autor, así como los artículos 35 y 36 fracción II de la Ley de la Universidad Autónoma del Estado de México; manifiesto mi autoría y originalidad de la obra mencionada que se presentó en Facultad de Ingeniería (lugar) C.U., Toluca, Edo. Méx para ser evaluada con el fin de obtener el Título Profesional de Ingeniero en Computación.

Así mismo expreso mi conformidad de ceder los derechos de reproducción, difusión y circulación de esta obra, en forma NO EXCLUSIVA, a la Universidad Autónoma del Estado de México; se podrá realizar a nivel nacional e internacional, de manera parcial o total a través de cualquier medio de información que sea susceptible para ello, en una o varias ocasiones, así como en cualquier soporte documental, todo ello siempre y cuando sus fines sean académicos, humanísticos, tecnológicos, históricos, artísticos, sociales, científicos u otra manifestación de la cultura.

Entendiendo que dicha cesión no genera obligación alguna para la Universidad Autónoma del Estado de México y que podrá o no ejercer los derechos cedidos.

Por lo que el autor da su consentimiento para la publicación de su trabajo escrito de evaluación profesional.

a) Texto completo.

b) Por capítulos.

c) Solamente portada y tabla de contenido.

Se firma presente en la ciudad de Toluca, a los 19 días del mes de Junio de 2015.

Victor Hugo Solis Ramos

Nombre y firma de conformidad