

# APLICACIÓN PARA VISUALIZAR INFORMACIÓN Y UBICACIÓN DE TODOS LOS CENTROS EDUCATIVOS Y DE ENSEÑANZA EN LA CIUDAD DE LOJA

Jhonson Pardo, Javier Fierro

*Departamento de Ciencias de la Computación y Electrónica, Universidad Técnica Particular De Loja  
San Cayetano Alto, Loja-Ecuador*

---

## Abstract

En el presente trabajo se habla sobre el desarrollo de una aplicación web dirigida a los centros educativos de la provincia de Loja. Actualmente existen aplicaciones con cierta similitud al proyecto pero se ha tratado de darle mayor funcionalidad integrando diferentes módulos que abastezcan a ciertas necesidades por parte de los usuarios.

Se ha utilizado diferentes herramientas como un gestor de base de datos para la carga de información, Mysql Workbench para el modelado, Sublime Text como editor de texto, etc.

Los protocolos que se han utilizado para el envío de información son HTTP, TCP/IP, además se explica cómo es el flujo de datos desde el primer momento en el que el usuario realiza una petición al servidor hasta obtener resultados.

Se ha utilizado arquitectura en capas debido a que la aplicación va enfocada al ámbito social y las consultas que se realizan son constantes. Además se ha implementado un método de encriptación de contraseñas denominado MD5 para una mayor seguridad de la aplicación en cuanto a los centros educativos.

Para que la aplicación esté disponible a los la subida del diferentes usuarios se ha adquirido un dominio y un host (BIZ.NF) y el que se ha subido el proyecto para la disponibilidad de las personas interesadas.

*Key words:* OSI; HTTP; TCP/IP; MySql Worbech; Dominio; Host; IP ; UTPL ; BDD

---

## 1 Introduction

Actualmente para conocer información de instituciones nos vemos obligados de dirigirnos a la institución, buscar a la secretaría y preguntar sobre todas las inquietudes, esto hace que tengamos que disponer de algún tiempo y conocer la ubicación de la institución. Por otro lado hay portales web que tienen ya información sobre instituciones educativas, pero con información muy limitada y no interactiva para el usuario y en mucho de los casos desactualizada.

Ante esta situación, la aplicación que proponemos dará solución a la falta de información estructurada y conjunta de todos los centros educativos existentes en la ciudad de Loja tales como: educación inicial, escuelas, colegios, institutos tecnológicos, universidades y academias.

Por otro lado el ministerio de educación y el sistema nacional de información a través de sus portales web ya ofrece información sobre los centros educativos pero

en un formato que a los usuarios no les llama la atención (xlsx). Es por eso que hemos propuesto una aplicación donde se pueda visualizar la ubicación exacta de los centros a través de la API de Google Maps y también poder dar a conocer información acerca de las instituciones a través de diagramas (Pastel, Barras).

Un problema que existe en una aplicación de este tipo es mantener la información actualizada, es por ello que la aplicación cuenta con cuentas para cada institución para que sean ellos mismos los que se encarguen de gestionar su información y ante un posible cambio la puedan actualizar de inmediato.

Entonces se puede finalizar manifestando que dicha aplicación está disponible para público en general con acceso a Internet, ya que se encuentra subida en un servidor en la nube, por otro lado también es importante tener en claro que el tener información completa, es vital para mantener informado al público, algo con que algunas aplicaciones no cuentan.

## 2 Trabajos Relacionados

En el medio local y nacional existen algunas páginas web que brindan información acerca de las instituciones educativas que existen en la ciudad de Loja.

Existen aplicaciones que muestran información de algunas instituciones educativas (Portal del Colegio (Colegio, 2010)) (Top Colegios (Colegios, 2012)) (Centros Educativos de la Ciudad de Loja (Colegio, 2010)) y las mismas dan a conocer solo el nombre de la institución y la dirección y en muchos de los casos esta información está desactualizada. Otra aplicación usa la api de google maps pero no trabaja bien ya que muestra la misma dirección en todos los centros (Tiching (Tiching, 2011)). Así mismo en el ámbito local existe una página web que muestra una información estructurada acerca de algunas instituciones de educación superior pero no es completa (Cursos y Carreras (Carreras, 2014)). Además en la página web del “Sistema Nacional de Información” a través de una aplicación muestra información de todos los centros educativos de Loja y es visible a través de un mapa pero solo nos muestra información de la ubicación y las coordenadas de las instituciones educativas y no muestra ninguna información relacionada con academias, o información más específica de los centros educativos (Sistema Nacional de Información (Información, 2015)). El proyecto presentado que a diferencia de los mencionados anteriormente, la aplicación muestra información completa que ha sido extraída del INEC, SNI y el ministerio de educación, además se ha trabajado con el API de Google Maps que muestra la ubicación de una manera personalizada para cada institución, también se usa el API de Highcharts que sirve exclusivamente presentar gráficas estadísticas sobre las instituciones. La aplicación cuenta con un buscador para que el usuario pueda acceder con facilidad a la institución que desea. Todos estos aspectos hace que la aplicación presentada sea a diferencia de las demás una integración de diferentes funcionalidades que favorece a los usuarios. También la información será actualizada, para ello tendremos cuentas de usuarios para cada institución que se encargará de actualizar su propia información conforme vaya cambiando. Dicha aplicación

## 3 Marco Teórico

### 3.1 Herramientas a utilizar

Para la aplicación se utilizó herramientas conocidas por su desempeño (Python, PostgreSQL). El manejo de cada una de las herramientas está enfocado a diferentes ámbitos, es decir, para estructurar los datos se utilizaron las herramientas: Phppgadmin, PostgreSQL, MySQL

**Table 1**  
**Herramientas utilizadas**

Nombre	Descripción
Django	Framework para desarrollo
PostgreSQL	Gestor de BDD
MySQL Workbench	Modelado de la BDD
Google Maps	API muestra ubicaciones
Highcharts	API estadísticas (pasteles)(barras)
HTML5	Diseño de la aplicación
CSS	Estilos para la aplicación
GitLab	Versionamiento
SublimeText	Codificación
Phppgadmin	Gestor de base de datos

Workbench. Para la maquetación de la página se utilizó HTML5 y CSS, para versionamientos del proyecto se utilizó Git Lab, etc.

### 3.2 Datos Estadísticos sobre las instituciones

La cantidad de instituciones educativas en la ciudad de Loja es mayor en los centros Fiscales y más aún en las escuelas con un total de 176 instituciones.

**Table 2**  
**Número de instituciones educativas**

Tipo de Institución	Categoría	Nro de Centros
Inicial	Fiscal	28
Inicial	Fiscomisional	2
Inicial	Particular Laico	13
Escuela	Fiscal	176
Escuela	Fiscomisional	2
Escuela	Municipal	3
Escuela	Particular Laico	18
Escuela	Particular Religioso	2
Colegio	Fiscal	32
Colegio	Fiscomisional	12
Colegio	Municipal	5
Colegio	Particular Laico	15
Colegio	Particular Religioso	6

Así como se presenta en la tabla anterior, los centros educativos de la ciudad de Loja están dominados por Escuelas, por lo que los usuarios que predominaran la aplicación web son los administradores de las escuelas.

**Table 3**  
**Cantidad de Instituciones por tipo**

Tipo de Institución	Nro de Centros
Inicial	43
Escuela	201
Colegio	70
Universidad	1
Instituto Tecnológico	1
Academias de Idiomas	4
Academias de Música	4
Academias de Deportes	1
Educación Especial	2
Centros de Rehabilitación	2
Centros de Audición y Lenguaje	1
Centros de formación Artesanal	1

En la tabla 3 se presenta el número de instituciones categorizados por el tipo de institución.

La información de cada institución está alojada en el servidor y disponible para el público.

### 3.3 Protocolos

Los protocolos que se han utilizado para nuestra aplicación son: TCP (Transmission Control Protocol), IP (Internet Protocol), HTTP (Hypertext Transfer Protocol).

Se utilizó el protocolo TCP / IP para la transferencia de los datos entre nodos distintos. (Yoon, 2009)

Los datos que son enviados mediante la aplicación por parte del usuario a la red (Actualización de los centros educativo) recorren la pila del protocolo TCP/IP desde la capa más alta de aplicación hasta la más baja de acceso a red. Cuando son recibidos, recorren la pila de protocolo en el sentido contrario. (Philippe ATELIN, 2014)

En cuanto al protocolo HTTP, en vista de que nuestra aplicación necesita de la WWW (World Wide Web) el intercambio de información se debe dar mediante dicho protocolo ya que hace posible la comunicación entre cliente/servidor y se basa en sencillas operación de solicitud/respuesta. (Li, 2013)

### 3.4 Seguridad

Cuando en una aplicación web se manejan cuentas de

usuarios puede resultar fácil para los hackers encontrar fallos a través de las mismas y realizar diversos daños a la aplicación y es por ello que es aconsejable usar el algoritmo MD5.

Es un algoritmo que transforma los datos a un nivel que es casi imposible tener acceso, es decir, nos brinda integridad de los datos. El algoritmo MD5 se lo representa con un número hexadecimal de 32 dígitos, dicho identificador se lo denomina hash. Un ejemplo de encriptación en un campo vacío sería:

MD5("") = d41d8cd98f00b204e9800998ecf8427eg

### 3.5 Flujo de datos en la Red

Aquí explicaremos cómo se realiza la comunicación entre nuestros cliente y servidor de nuestra aplicación a través del Protocolo TCP/IP y que sucede en cada una de sus capas.

#### Acceso a la red:

Al hacer clic en un link (pagina web) iniciamos un flujo de información dentro del computador y esta información viaja hacia la mensajería personal donde el protocolo IP lo empaqueta, etiqueta y pone en camino . Cada paquete es limitado en su tamaño el local de mensajería debe decidir cómo dividir la información y como empaquetarla . Cada paquete necesita una etiqueta donde se describa información importante tales como: La Dirección del remitente (Cliente-Servidor), del destinatario (Cliente-Servidor) y el tipo de paquete que es y debido a que este paquete viajará por internet también recibe una etiqueta para el servidor proxy . En este punto los datos abandona nuestra máquina y salen hacia la red cableada de nuestra corporación, el paquete es lanzado hacia la red de área local (LAN). La red LAN es nada controlada y encontramos diversos tipos de paquetes. El router local lee las direcciones de los paquetes y si es necesario pone a los paquetes en otra red. El router es el símbolo de control entre una red sin la misma. Cuando los paquete dejan el ruteador siguen su camino a través de la intranet o red corporativa. Más adelante nos encontramos con el switch ruteador que trabaja rápido y suelta los paquetes enrutando los hábilmente por su camino parecido a un pinball. Cuando los paquetes llegan a su destino son recolectados por la interfaz de red para ser enviados al siguiente nivel en este caso el proxy El proxy es usado como intermediario con la función de establecer y compartir entre varios usuarios una unica conexion de internet y también por razones de seguridad. El proxy abre el paquete y busca la dirección web o url y dependiendo si la dirección es admisible el paquete se enviará hacia internet. (En caso de que existan direcciones que no cuenten con la aprobación del proxy según como haya sido configurado de acuerdo a políticas corporativas o de administración y tales direcciones son inmediatamente destruidas.) Luego de que nuestro paquete fue revisado por el proxy, vuelve a la ruta dentro de nuestra LAN. El próximo paso es cruzar el firewall corporativo o local que sirve para prevenir

intromisiones indeseables provenientes de internet y evita que la información delicada de nuestra empresa sea enviada a internet, una vez que pasa que se pasa el firewall un router recoge cada paquete y lo coloca en un camino o ancho de banda (en caso de que un paquete nuestro no haya logrado recorrer todo ese camino,, cuando el protocolo IP no obtiene un recibo de que el paquete fue recibido en el tiempo debido simplemente envía un paquete de reemplazo )y estamos listos para pasar a la siguiente capa del protocolo.

Internet

Aquí no encontramos con una telaraña de redes interconectadas la cual se extiende por todo el orbe routers y switches establecen enlaces entre las redes. Las rutas o caminos que los paquetes pueden tomar son: satélites, cables telefónicos o cables transoceánicos y no siempre toman el camino más corto pero siempre llegan a su destino (cliente-servidor). Cerca de llegar al fin de nuestro viaje y encontrado al computador o servidor donde está alojada la página web solicitada encontraremos otro firewall . El firewall está diseñado para dejar entrar a aquellos paquetes que cumplen con el criterio de selección y este firewall tiene abierto lo puertos 80 y 25 que son las puertas de entrada y todo intento en los demás puertos será cancelado. El puerto 25 es usado para paquetes de correo, mientras que el puerto 80 es la entrada de los paquetes de internet hacia el servidor web.

Transporte

Dentro del firewall los paquetes son filtrados más rigurosamente y para los paquetes que no fueron dudosos, pasan hacia la interfaz para ser llevados hasta el servidor Web, uno por uno lo paquetes son recibidos, abierto y desempacados. La información que contienen (nuestra solicitud de información) es enviada la aplicación del servidor web.

Aplicación:

El paquete en si es reciclado, listo para ser usado otra vez y llenado con la información solicitada, etiquetado y enviado de regreso hacia nosotros. Regresa por el firewall del servidor web, router y a través de todo internet , devuelta a nuestro firewall corporativo o local y hasta nuestra interfaz en nuestro computador, aquí es suministrado el navegador de internet con la información solicitada.

4 Metodología

4.1 Diseño

4.1.1 Diagrama de flujo

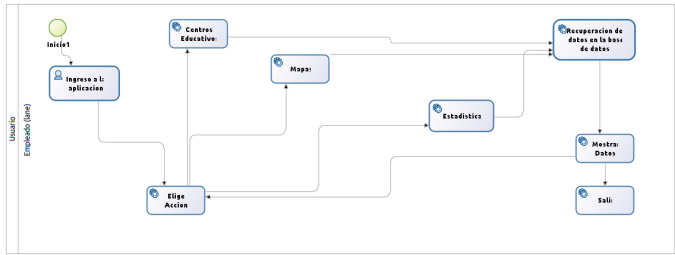


Fig. 1. Flujo de datos de la aplicación

En la figura **Figura 1** se aprecia el flujo de la aplicación para las personas que accedan a la página web. Primeramente el usuario deberá ingresar al portal web, una vez que haya ingresado deberá escoger el criterio de búsqueda de las instituciones educativas, ya que se cuenta con diferentes criterios y finalmente se pueda visualizar la información correspondiente.

En caso que el usuario sea un cliente de la aplicación (Administrador de la Institución) tendrá los privilegios para poder gestionar su información como por ejemplo la actualización de los datos.

4.1.2 Base De Datos

Para la implementación del sistema se realizó el modelado en la herramienta Mysql Workbench, la cual involucra información de las diferentes entidades

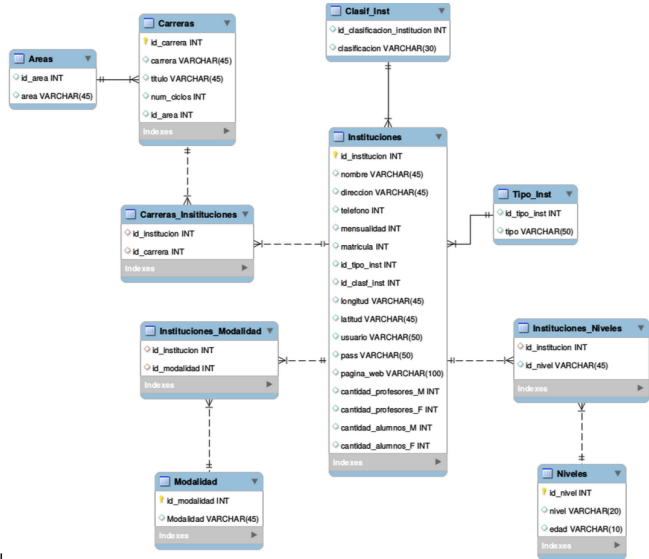


Fig. 2. Modelo de Base de Datos.

educativas de la ciudad de Loja. La información que abarca la aplicación web es entorno a los centros educativo, entre ellos tenemos:

Tipo de Institución educativa, como puede ser: Universidades, Escuelas, Colegios, Academias (Ingles, Futbol, etc.), centros tecnológicos y demás.

Dependiendo del tipo de centro que sea se maneja información diferente como por ejemplo las universidades tienen áreas y que carreras ofrecen cada una y la disponibilidad de modalidades.

Además se brinda información sobre la categoría del centro educativo como: Institución pública, privada, etc. Toda esta información se encuentra estructurada como muestra la **Figura 2**, es decir, cada tabla se encuentra relacionada entre sí para poder brindar una estructura compacta de datos con respecto a los centros educativos.

#### 4.1.3 Aplicación

Una vez obtenido el modelo se trabajó con HTML (Lenguaje de Marcas de Hipertexto) para el respectivo diseño de la página junto con CSS (Hojas de Estilo) para poder obtener un mejor formato y presentación de la aplicación. El lenguaje de programación con el cual se trabajó fue python, utilizando el framework de Django.

Para el diseño de la aplicación se trabajó con la arquitectura Cliente – Servidor debido a que la aplicación va enfocado al ámbito social y esta arquitectura nos ofrece la usabilidad y flexibilidad que se necesita.

Debido a que se opera directamente con la base de datos se ha implementado seguridad de encriptación de contraseñas para no ser víctima de un ataque por agentes externos. El método de encriptación utilizado es MD5.

Además se utilizó la arquitectura en capas (**Figura 3**) para una mejor respuesta entre el cliente y el servidor.

#### 4.2 Arquitectura en Capas

La arquitectura que se ha decidido utilizar para el proyecto es arquitectura en capas debido a que las demás arquitecturas no se adaptan a las necesidades que se busca por lo que presenta algunos problemas como la

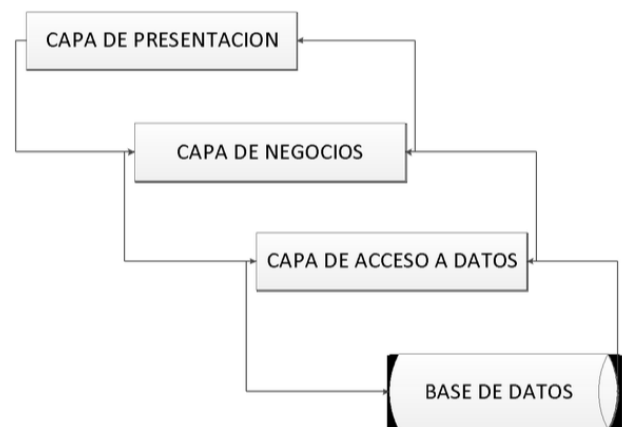


Fig. 3. Arquitectura En Capas.

inconsistencia de los datos que se intercambian entre el cliente y el servidor.

Es importante tener en cuenta que además de la arquitectura en capas se refleja la arquitectura cliente-servidor, no es obligatorio que el cliente y el servidor residan en la misma máquina; en la práctica, resulta bastante común situar el servidor en una determinada ubicación dentro de una red de área local y los clientes en las ubicaciones restantes. (Thomas, 2014).

En el proyecto propuesto el cliente son todas las entidades como las escuelas, colegios, universidades, etc. que interactúan como usuario directo con el sistema ya que por medio de un navegador web se accede al sistema para la gestión de la información en el portal, y así mismo cualquier usuario que interactúe con la página. El papel del servidor lo realiza nuestro host (<http://www.biz.nf>) en donde se aloja toda la información que se ha recolectado e informacion que las entidades han subido.

### 5 Resultados

#### 5.1 Base de Datos

Se modeló la base de datos, utilizando lenguaje sql se crearon las tablas y se insertó datos, dando como resultado el script de la base de datos que fue implementada en el programa phpPgadmin (**Figura 4**) para luego poder subirla al host.

Tabla	Dueto	Tablaspace	Cantidad de filas	Acciones									
<input type="checkbox"/> areas	academias		4	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> auth_group	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> auth_group_permissions	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> auth_permission	academias		54	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> auth_user	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> auth_user_groups	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> auth_user_user_permissions	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> carreras	academias		34	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> clasificacion_institucion	academias		5	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> django_admin_log	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> django_content_type	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> django_migrations	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> django_session	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> instituciones	academias		331	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> instituciones_carreras	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> instituciones_jornada	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> instituciones_modidades	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> instituciones_niveles	academias		0	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> jornada	academias		2	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> modalidades	academias		2	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> niveles	academias		19	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	
<input type="checkbox"/> tipo_institucion	academias		13	Examinar	Seleccionar	Insertar	Vaciar	Modificar	Eliminar	Limpiar	Analizar	Actualizar indices	

Fig. 4. Base de Datos Estructurada.

## 5.2 Programación

En cuanto a programación fue desarrollado en python con el framework django y las lineas de codigo fueron desarrolladas con el editor de textos sublime text (Figura 5 ).

```

models.py
44 class Instituciones(models.Model):
45     id_institucion = models.IntegerField(primary_key=True)
46     nombre = models.CharField(max_length=200, blank=True)
47     direccion = models.CharField(max_length=250, blank=True)
48     telefono = models.IntegerField(blank=True, null=True)
49     mensualidad = models.DecimalField(max_digits=65535, decimal_places=65535, blank=True, null=True)
50     matricula = models.DecimalField(max_digits=65535, decimal_places=65535, blank=True, null=True)
51     id_tipo_institucion = models.ForeignKey('TipoInstitucion', db_column='id_tipo_institucion', blank=True, null=True)
52     id_clasificacion_institucion = models.ForeignKey('ClasificacionInstitucion', db_column='id_clasificacion_institucion', blank=True, null=True)
53     latitud = models.CharField(max_length=20, blank=True)
54     longitud = models.CharField(max_length=20, blank=True)
55     usuario = models.CharField(max_length=50, blank=True)
56     pass_field = models.CharField(db_column='pass', max_length=50, blank=True) # Field renamed because it
57     pagina_web = models.CharField(max_length=100, blank=True)
58     cant_profesores = models.IntegerField(blank=True, null=True)
59     cant_profesores_f = models.IntegerField(blank=True, null=True)
60     cant_alumnos = models.IntegerField(blank=True, null=True)
61     cant_alumnos_f = models.IntegerField(blank=True, null=True)
62
63     class Meta:
64         managed = False
65         db_table = 'instituciones'
66
67
68 class InstitucionesCarreras(models.Model):
69     id_institucion = models.ForeignKey(Instituciones, db_column='id_institucion', blank=True, null=True)
70     id_carrera = models.ForeignKey(Carreras, db_column='id_carrera', blank=True, null=True)
71
72     class Meta:
73         managed = False
74         db_table = 'instituciones_carreras'
75
76
77 class InstitucionesJornada(models.Model):
78

```

Fig. 5. Código en Django.

Por otro lado existe la necesidad de que la aplicación web pueda ser visualizada desde cualquier dispositivo y para ello fue necesario insertar la librería js. (Figura 6 ) , la misma que permite que todas las etiquetas html existentes en la aplicación web sean consumidas desde cualquier tipo de dispositivo con acceso a la red.

Nombre	Fecha de modifica...	Tipo	Tamaño
.DS_Store	18/11/2015 0:08	Archivo DS_STORE	7 KB
modernizr.custom.js	18/11/2015 0:08	Archivo JavaScript	9 KB

Fig. 6. Librería Jsn.

En cuanto a maquetado de la página se utilizó el framework Bootstrap ((Figura 7 ))que es de útil ayuda al diseño de aplicaciones web, en conjunto con el lenguaje de etiquetado html y css, se obtuvo el diseño de la página final.

Nombre	Fecha de modifica...	Tipo	Tamaño
bootstrap	18/11/2015 0:18	Carpeta de archivos	
book.css	18/11/2015 0:08	Documento de ho...	10 KB
demo.css	18/11/2015 0:08	Documento de ho...	4 KB
normalize.css	18/11/2015 0:08	Documento de ho...	2 KB
reset.css	18/11/2015 0:08	Documento de ho...	1 KB
style.css	18/11/2015 0:08	Documento de ho...	2 KB
styleE.css	18/11/2015 0:08	Documento de ho...	1 KB

Fig. 7. Archivos Bootstrap y CSS.

Finalmente para visualizar las ubicaciones de los centros educativos y las academias se utilizó la api de google ((Figura 8 )), donde se insertó mediante un script en un archivo, la función del mapa y las coordenadas del lugar (previamente recuperadas de la base de datos).

```

<script src="https://maps.googleapis.com/maps/api/js?v=3.exp"></script>
<script>
function initialize() {
var myLatLng = new google.maps.LatLng(-4.033556, -79.202718);
var mapOptions = {
zoom: 12,
center: myLatLng
}
var map = new google.maps.Map(document.getElementById('map-canvas'),
mapOptions);

{% for lugar in instituciones %}
var marcador = new google.maps.LatLng({{lugar.latitud}}, {{lugar.
longitud}});
var marker = new google.maps.Marker({
position: marcador,
map: map,
title: '{{lugar.nombre}}'
});
google.maps.event.addListener(marker, 'click', function(){
var popup = new google.maps.InfoWindow();

```

Fig. 8. Api de Google Maps.

## 5.3 Implementación de Seguridad

Debido a que la información debe ser correcta siempre, fue necesario implementar encriptación a las contraseñas de cada cuenta y la misma se la realizó aplicando el siguiente código de encriptación en la aplicación. (Figura 9 )

```

$password = md5($password);

```

Fig. 9. Código de Encriptación MD5.

## 5.4 Subir Aplicación al host

Primero ingresamos al sitio web del host y seleccionamos la operación que deseamos realizar, para poder subir los archivos de la aplicación web damos en click en file manager, donde podemos gestionar y cargar todos los archivos que se han programado, y que luego serán visualizados a través del dominio: http://centros-educativos-loja.co.nf/ El host ofrece una interfaz web similar a phpmyadmin para poder gestionar la base de datos y dándonos el puerto 3306 para poder

Database details for 1996111_centros	
Below you find detailed information of your database	
Database Host	fdb6.biz.nf
Database Port	3306
Database Name	1996111_centros
Database User	1996111_centros
Database Password	The password you have chosen <a href="#">change</a>
Database Version	5.5

Fig. 10. Características de la BDD en el host.

cargarla, se puede observar todos los datos en la (Figura 10) .

### 5.5 Aplicación Web Online

Una vez subido los archivos programados y la base de datos al host podemos visualizar la aplicación ya en línea. En la (Figura 11) se encuentra la página de Inicio: En la figura (Figura 12) se observa la estructura que



Fig. 11. Página de Inicio de la Aplicación.

tiene cada institución en la aplicación :

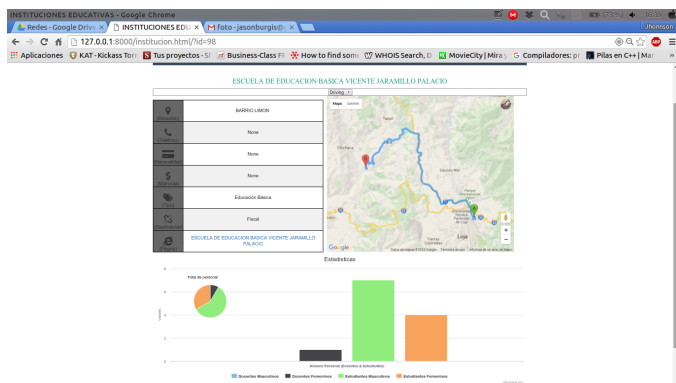


Fig. 12. Visualización de Centros en la Aplicación.

Finalmente tenemos las siguientes características de red de la Aplicación Web :

- Dominio: <http://centros-educativos-loja.co.nf/>
- Host: biz.nf
- Dirección IP: 83.125.22.197
- Puerto que el servidor nos da para la comunicación cliente servidor: 80
- Puerto de Base de Datos: 3306

## 6 Conclusiones

- Al momento de que un usuario interactúa con la aplicación, está trabajando con la dirección IP del sitio que nos dio el host y realiza lo que hay en cada capa del protocolo TCP/IP
- Mediante el dominio y el protocolo HTTP los usuarios pueden encontrar y hacer uso de la aplicación web. <http://centros-educativos-loja.co.nf/>
- El servidor del host nos da como puerta de entrada para la comunicación cliente servidor el puerto 80, y para la interacción con la base de datos tenemos el puerto 3306.
- El uso de framework reduce el tiempo notablemente, además de un código mayormente optimizado, más ordenado y entendible por el simple hecho de trabajar con un patrón denominado Modelo Vista Controlador.
- Al momento en que el cliente realiza una petición al servidor los datos recorren cierta trayectoria por la que puede ser propenso a robo de información, por ese motivo es necesario implementar algún tipo de seguridad como la encriptación de datos, el cifrado, etc. para no ser víctima de dicha catástrofe.

## References

- Carreras, L. (2014). *Centros educativos loja*. Retrieved from <http://www.cursosycarreras.com.ec/instituciones-educativas-ecuador>
- Colegio, P. (2010). *Colegios de loja*. Retrieved from [http://www.portaldelcolegio.com/colegios/colegios\\_loja.php](http://www.portaldelcolegio.com/colegios/colegios_loja.php)
- Colegios, T. (2012). *Colegios privados de loja*. Retrieved from <http://www.topcolegios.com/directorio-colegios-privados/ecuador/loja>
- Información, S. (2015). *Centros educativos loja*. Retrieved from <http://sni.gob.ec/inicio>
- Li, Z. (2013). Http-ccn gateway: Adapting http protocol to content centric network.
- Philippe ATELIN, J. D. (2014). *Tcp/ip y protocolos de internet*. New York: TechNote coleccion.
- Thomas, C. (2014). *Sistema de base de datos un enfoque práctico para el diseño, implementación y gestión*. London: PEARSON.
- Tiching. (2011). *Colegios en loja*. Retrieved from <http://ec.tiching.com/colegios/loja/52047>
- Yoon, I.-S. (2009). Implementation of lightweight tcp/ip for small, wireless embedded systems.