

# ARQUITECTURAS WEB

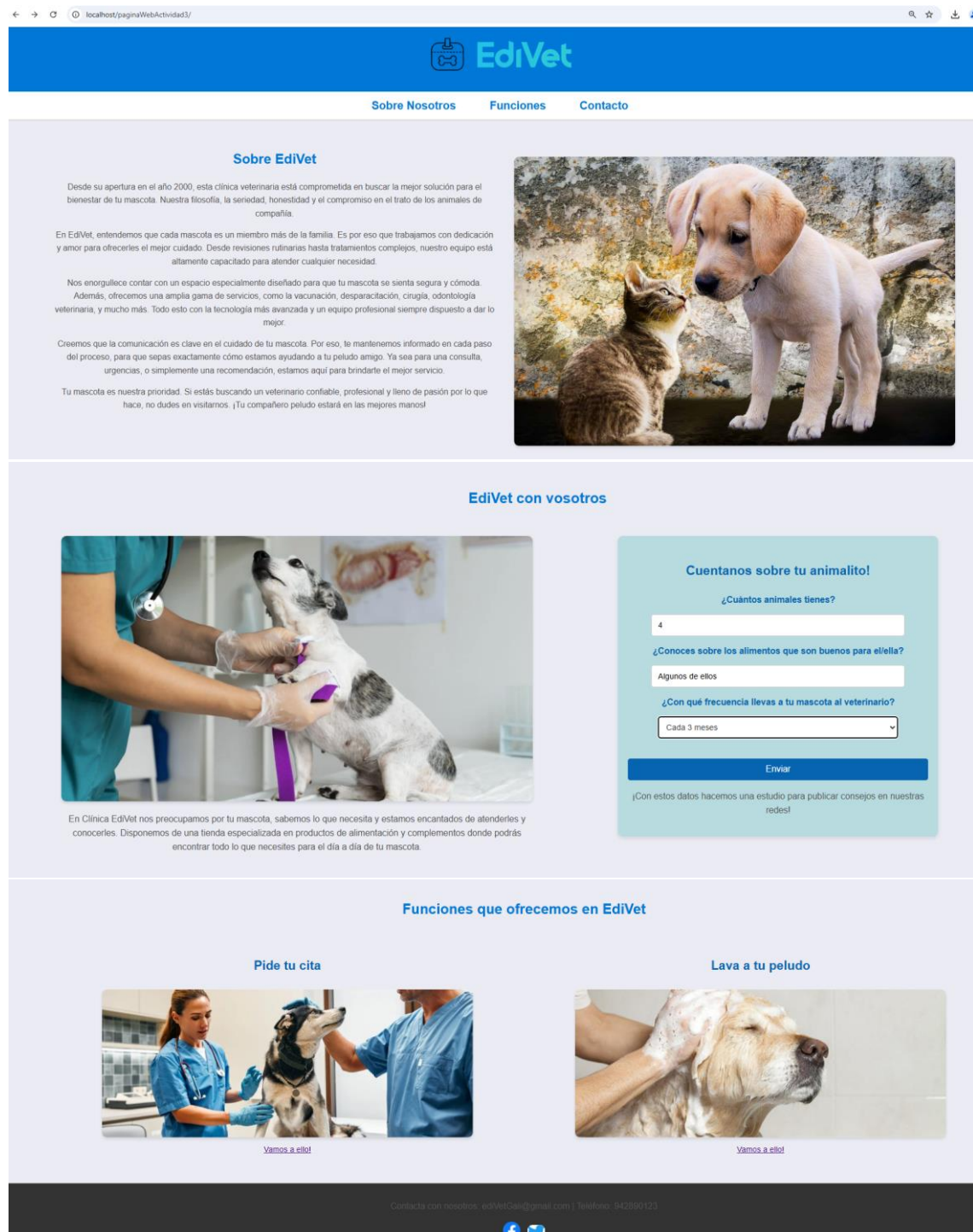
Edgar Núñez Arana

# Tabla de Contenido

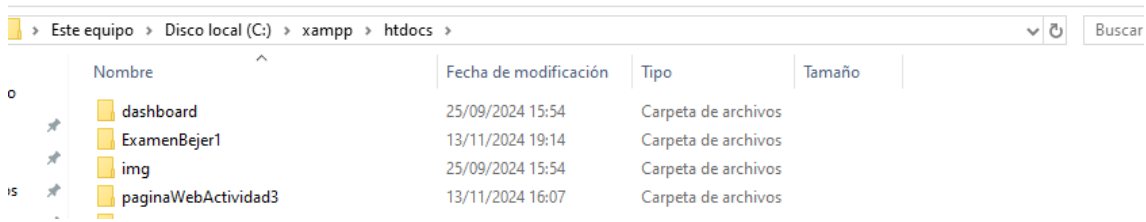
<b>Explicación Pagina Web .....</b>	<b>2</b>
<b>Página Web Estática: .....</b>	<b>2</b>
<b>Funciones Aplicación Web .....</b>	<b>3</b>
<b>Conexión entre ambas .....</b>	<b>6</b>
 <b>Investigación .....</b>	 <b>8</b>
<b>¿Cuál fue la parte más compleja de esta actividad? ¿Cómo     lograste resolverla? .....</b>	 <b>8</b>
<b>¿Puedes investigar acerca de otro servidor web y compararlo     con Apache? .....</b>	 <b>8</b>

## Explicación Pagina Web

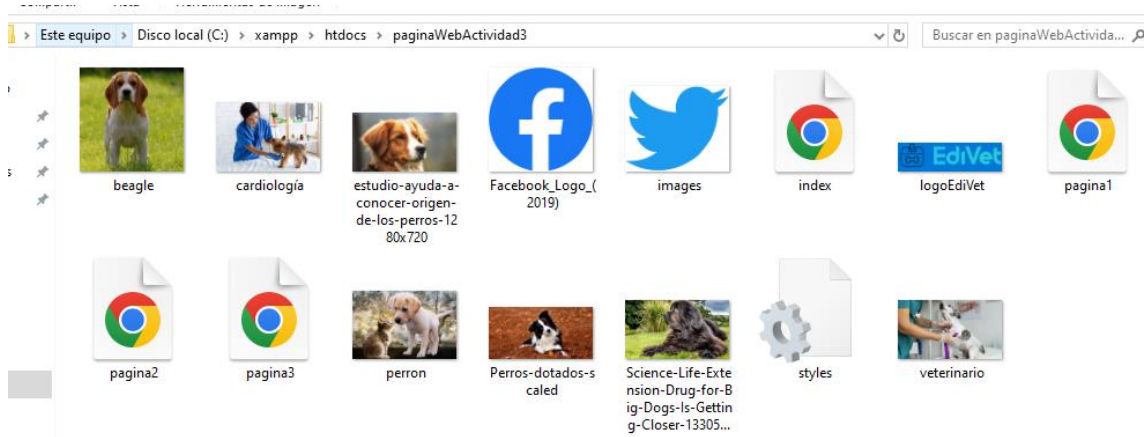
### Página Web Estática:



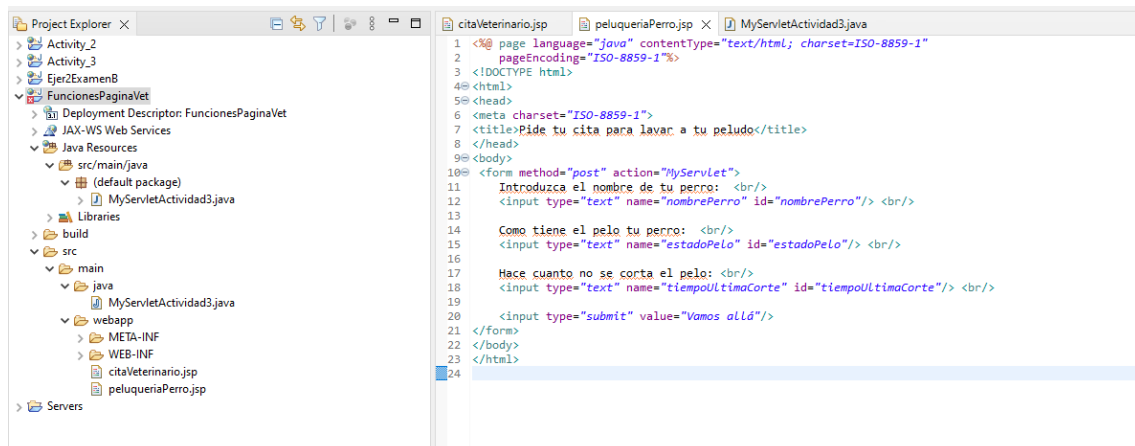
Primeramente creamos nuestra página estática la cual esta levantada desde Apache para ello guardaremos la carpeta con la propia página creada(index,css, imágenes...) en la ruta httdocs.



Como se puede ver nuestra página está en la carpeta paginaWebActividad3 la cual se encuentran nuestros archivos previamente explicados



## Funciones Aplicación Web



Con eclipse crearemos una Proyecto dynamic web el cual nos servirá para implementar nuestras funciones para ello usaremos servlets y jsp en mi caso creo 2 archivos jsp para las 2 funciones que tiene mi página web.

```

citaVeterinario.jsp      peluqueriaPerro.jsp  MyServletActividad3.java
1  <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2    pageEncoding="ISO-8859-1"%>
3  <!DOCTYPE html>
4  <html>
5  <head>
6    <meta charset="ISO-8859-1">
7    <title>Pide tu cita para lavar a tu peludo</title>
8  </head>
9  <body>
10 <form method="post" action="MyServlet">
11   Introduzca el nombre de tu perro: <br/>
12   <input type="text" name="nombrePerro" id="nombrePerro"/> <br/>
13
14   Como tiene el pelo tu perro: <br/>
15   <input type="text" name="estadoPelo" id="estadoPelo"/> <br/>
16
17   Hace cuanto no se corta el pelo: <br/>
18   <input type="text" name="tiempoUltimaCorte" id="tiempoUltimaCorte"/> <br/>
19
20   <input type="submit" value="Vamos allá"/>
21 </form>
22 </body>
23 </html>
24
citaVeterinario.jsp  peluqueriaPerro.jsp  MyServletActividad3.java
1  <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2    pageEncoding="ISO-8859-1"%>
3  <!DOCTYPE html>
4  <html>
5  <head>
6    <meta charset="ISO-8859-1">
7    <title>Pide tu cita para revisar a tu peludito</title>
8  </head>
9  <body>
10 <form method="post" action="MyServlet">
11   Introduzca el nombre de tu perro: <br/>
12   <input type="text" name="nombrePerro" id="nombrePerro"/> <br/>
13
14   Cuanto pesa tu animalito: <br/>
15   <input type="text" name="pesoAnimal" id="pesoAnimal"/> <br/>
16
17   Que le pasa a tu animalito: <br/>
18   <input type="text" name="problemaAnimal" id="problemaAnimal"/> <br/>
19
20   <input type="submit" value="Vamos allá"/>
21 </form>
22 </body>
23 </html>

```

En este caso son 2 formularios para nuestra veterinario

```

citaVeterinario.jsp  peluqueriaPerro.jsp  *MyServletActividad3.java X
3  import javax.servlet.ServletException;
4  import javax.servlet.annotation.WebServlet;
5  import javax.servlet.http.HttpServlet;
6  import javax.servlet.http.HttpServletRequest;
7  import javax.servlet.http.HttpServletResponse;
8
9  @WebServlet("/MyServlet")
10 public class MyServletActividad3 extends HttpServlet {
11     private static final long serialVersionUID = 1L;
12
13     // Constructor
14     public MyServletActividad3() {
15         super();
16     }
17
18     protected void doGet(HttpServletRequest request, HttpServletResponse response)
19         throws ServletException, IOException {
20         response.setContentType("text/html");
21         PrintWriter out = response.getWriter();
22         out.println("<h1>Bienvenido a la página de citas</h1>");
23     }
24
25     protected void doPost(HttpServletRequest request, HttpServletResponse response)
26         throws ServletException, IOException {
27         //vemos que form es (citaVeterinario.jsp o peluqueriaPerro.jsp)
28         String action = request.getParameter("action");
29         if (action == null) {
30             action = "revisar";
31         }
32
33
34         String nombrePerro = request.getParameter("nombrePerro");
35         String mensajeRespuesta = "";
36
37         if ("lavar".equals(action)) {
38             String estadoPelo = request.getParameter("estadoPelo");
39             String tiempoUltimaCorte = request.getParameter("tiempoUltimaCorte");
40
41             mensajeRespuesta = "<h1>Cita para Lavar al Perro</h1>";
42             mensajeRespuesta += "<p>Nombre del perro: " + nombrePerro + "</p>";
43             mensajeRespuesta += "<p>Estado del pelo: " + estadoPelo + "</p>";
44             mensajeRespuesta += "<p>Último corte hace: " + tiempoUltimaCorte + "</p>";
45         } else if ("revisar".equals(action)) {
46             String pesoAnimal = request.getParameter("pesoAnimal");
47             String problemaAnimal = request.getParameter("problemaAnimal");
48
49             mensajeRespuesta = "<h1>Cita para Revisar al Animalito</h1>";
50             mensajeRespuesta += "<p>Nombre del perro: " + nombrePerro + "</p>";
51             mensajeRespuesta += "<p>Peso del animalito: " + pesoAnimal + "</p>";
52             mensajeRespuesta += "<p>Problema: " + problemaAnimal + "</p>";
53         }
54     }

```

Además tendremos que crear una clase java la cual servirá para dar la funcionalidad a nuestros forms (citaPeluqueria.jsp y peluqueriaPerro.jsp)

Para conectar a la funcionalidad tendremos que exportar nuestro archivo en formato .war y guardarlo en la carpeta de webapps en la carpeta tomcat.



Este equipo > Disco local (C:) > xampp > tomcat > webapps >			
	Nombre	Fecha de modificación	Tipo
ceso rápido	docs	25/09/2024 15:55	Carpeta de archivo:
scritorio	Ejer2ExamenB	13/11/2024 19:50	Carpeta de archivo:
escargas	examples	25/09/2024 15:55	Carpeta de archivo:
ocumentos	FuncionesPaginaVet	15/11/2024 15:37	Carpeta de archivo:
nágenes	host-manager	25/09/2024 15:55	Carpeta de archivo:
ctividad2	manager	25/09/2024 15:55	Carpeta de archivo:
xamen	ROOT	25/09/2024 15:55	Carpeta de archivo:
úsica	Ejer2ExamenB	13/11/2024 19:50	Archivo WAR
webapps	FuncionesPaginaVet	15/11/2024 15:37	Archivo WAR
eDrive			

## Conexión entre ambas

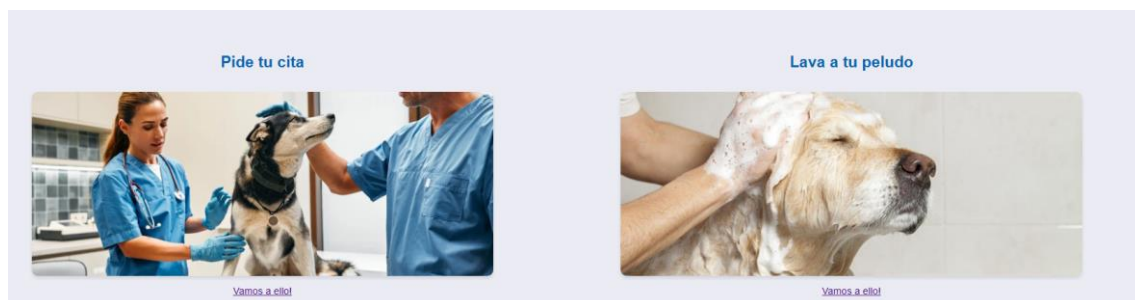
Además de guardar cada archivo en su directorio correspondiente tendremos que añadir una ruta a nuestra página estática.

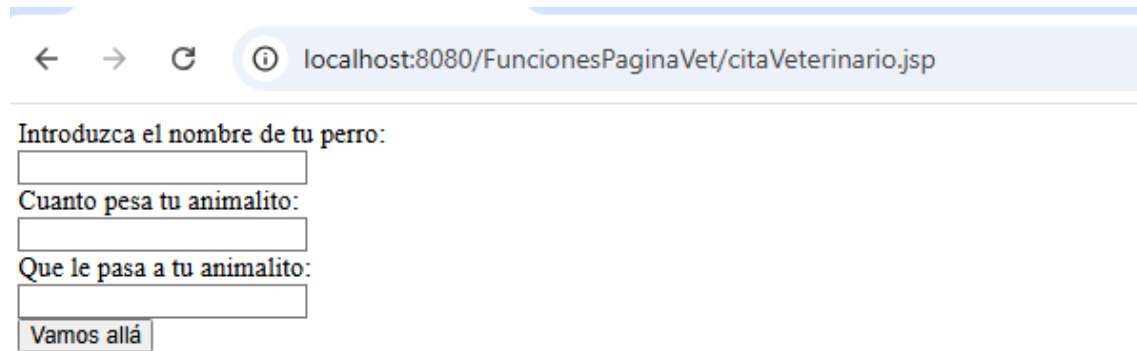
```

<h3>Funciones que ofrecemos en EdVet</h3>
<div class="grid-funciones">
  <div class="p1">
    <h4>Pide tu cita</h4>
    
    <a href="http://localhost:8080/FuncionesPaginaVet/citaVeterinario.jsp">Vamos a ello!</a>
  </div>
  <div class="p2">
    <h4>Lava a tu peludo</h4>
    
    <a href="http://localhost:8080/FuncionesPaginaVet/peluqueriaPerro.jsp">Vamos a ello!</a>
  </div>
</div>

```

Como se puede apreciar con este href (<http://localhost:8080/FuncionesVet/...>) podremos conectar a las funciones desde nuestra página estática.



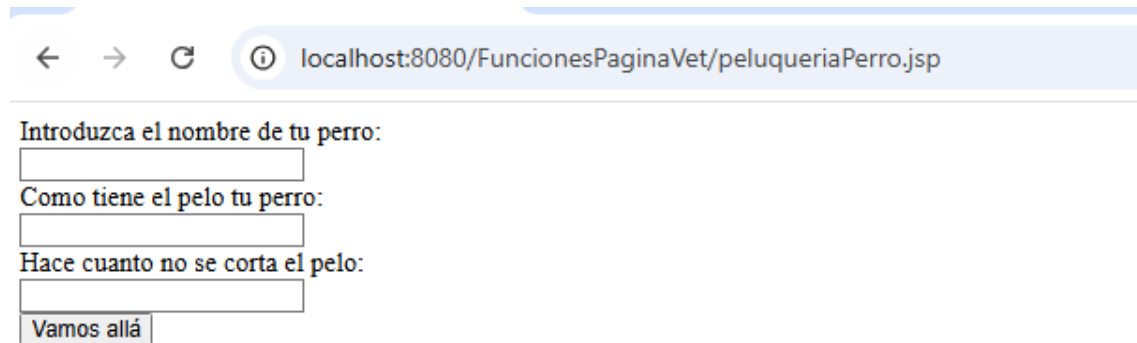


← → ↻ ⓘ localhost:8080/FuncionesPaginaVet/citaVeterinario.jsp

Introduzca el nombre de tu perro:

Cuanto pesa tu animalito:

Que le pasa a tu animalito:



← → ↻ ⓘ localhost:8080/FuncionesPaginaVet/peluqueriaPerro.jsp

Introduzca el nombre de tu perro:

Como tiene el pelo tu perro:

Hace cuanto no se corta el pelo:

Como se puede apreciar desde la ruta

[Vamos a ello!](#)

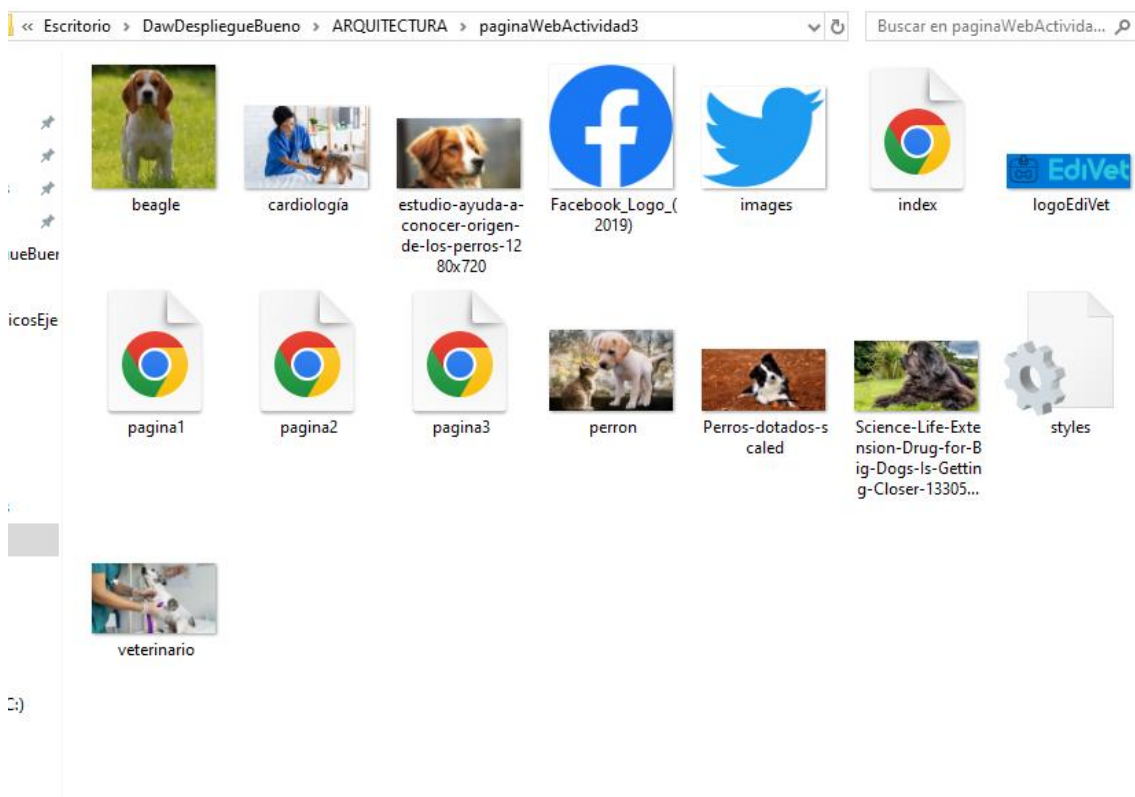
iremos a nuestras funciones previamente levantadas desde Tomcat.



## Investigación

### ¿Cuál fue la parte más compleja de esta actividad? ¿Cómo lograste resolverla?

La parte más compleja de la actividad para mí fue entender la lógica de que parte tendría que ir al “Tomcat” y al “Apache” además de como relacionar las funciones “jsp” en mi página estática ya que yo pensaba que podría implementar las funciones en nuestra página estática en vez de abrir una página nueva.



Como se puede ver había creado anteriormente paginas extras .html donde pensaba que podría introducir ahí la funcionalidad como si un archivo js se tratase, la solución fue comprender correctamente la actividad

### ¿Puedes investigar acerca de otro servidor web y compararlo con Apache?

Además del servidor web Apache hay otros más como IIS el cual está desarrollada por Microsoft por lo que nos daría un buen respaldo se puede administrar a través de internet y soporta ODBC y .NET framework y lenguaje ASPX pero tendría desventajas en frente de Apache como que su licencia no es gratuita, no es multiplataforma y es difícil controlar la dirección.