

# Un foro implementado en páginas ASP y base de datos Access.

Por Felipe Carrillo Oliva.

ELO-330 Programación de Sistemas.

2do. Semestre de 2002.

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

1

## Introducción

- ASP (Active Server Pages), permite crear páginas web dinámicas, permitiendo la interacción con el usuario.
- Los usuarios pueden tener acceso a la información que necesiten, mediante la implementación de aplicaciones ASP y conexiones a bases de datos.
- En esta presentación se muestra cómo se implementó una aplicación ASP, en lenguaje VBScript, para un foro electrónico, con la información guardada en un archivo de base de datos Access.
- Se verán esquemas que ilustran la implementación de la aplicación, así como algunas herramientas ASP utilizadas, y el diseño de la base de datos.

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

2

## Cómo se trabajó

- La aplicación ASP se programó en Microsoft Visual Interdev.
- El servidor que acoge la aplicación es Internet Information Server IIS.
- La base de datos se implementó en Microsoft Access.

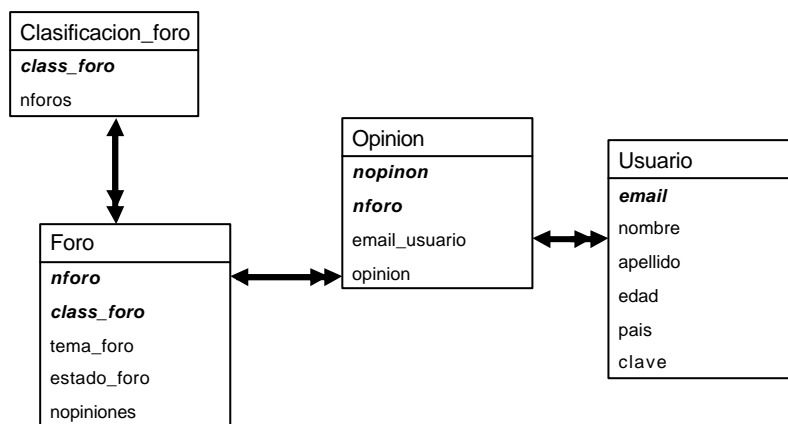
04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

3

## Diseño de la base de datos.

- Modelo entidad – relación.

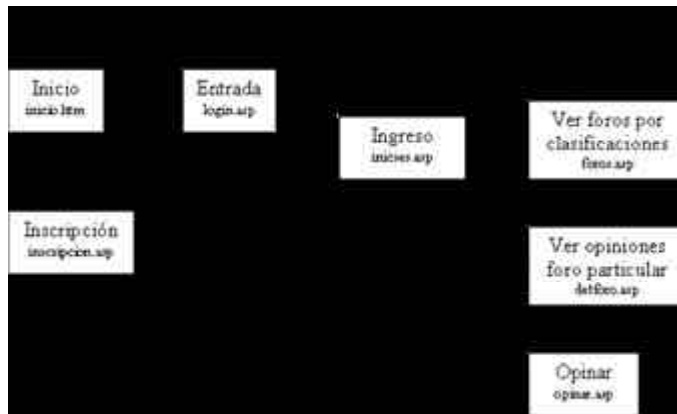


04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

4

# Esquema de la aplicación ASP.



04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

5

## Características ASP.

- **El objeto Request** permite recuperar la información enviada por un usuario al servidor.

Ejemplo

```
<FORM METHOD="Post" ACTION="login.asp" id=form1 name=form1 >  
  E-mail: <INPUT NAME="Email"><BR>  
  Clave: <INPUT TYPE="password" NAME="Clave"><BR>  
  <INPUT TYPE="Submit" VALUE="Enviar" id=Submit1 name=Submit1 >  
</FORM>
```

```
<%  
Clave1=Request.Form("Clave")  
Email=Request.Form("Email")  
%>
```

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

6

# Características ASP.

- La colección *QueryString* recupera los valores de las variables de cadena de consulta enviada en una petición HTTP.

Ejemplo:

```
<FORM METHOD="Post" ACTION="opinar.asp?clas=<%=clasforo%>&nro=<%=nroforo%>"
  id=form1 name=form1>
  <TEXTAREA rows=10 cols=60 id=textarea1 name="optext">
</TEXTAREA><BR>
  <INPUT TYPE="Submit" VALUE="Enviar" id=Submit1 name=Submit1>
</FORM>
```

En la página llamada por el formulario:

```
<%
nroforo=Request.QueryString("nro")
clasforo=Request.QueryString("clas")
%>
```

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

7

# Características ASP.

- La colección *Cookies* permite recuperar el valor de una cookie:

Ejemplo:

```
Email=Request.Cookies ("idusuario")("email")
```

- **El objeto Response** se usa para enviar al cliente cualquier clase de información, y para manejar la manera de su envío y su retorno.
  - El método *Redirect* indica al navegador que conecte con otra dirección URL.

Ejemplo:

```
Response.Redirect("iniceses.asp")
```

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

8

## Características ASP.

- La colección *Cookies* permite crear y modificar cookies en el navegador cliente.

Ejemplo:

```
Response.Cookies("idusuario")("email")=Email
```

## Características ASP.

- **El objeto Server**, permite consultar aspectos del servidor y trabajar directamente con sus características.

- El método *CreateObject* crea una instancia de un componente instalado en el servidor.

Ejemplo:

```
SET Obj_Conn = Server.CreateObject ("ADODB.Connection")
```

- El método *Open* realiza la conexión real a la base de datos.

Ejemplo: `Obj_Conn.Open "Foro"`

- Este ejemplo, muestra como se abre una conexión usando el DSN *Foro*, el cuál posee la información sobre la ubicación de la base de datos.

## Características ASP.

- El método *Close* cierra la conexión a la base de datos:

Ejemplo: `Obj_Conn.Close`

- **El objeto RecordSet** permite obtener acceso a las tablas de la base de datos:

Ejemplo:

`SET Obj_RSusuario = Server.CreateObject ("ADODB.RecordSet")`

- El método *Open* abre un conjunto de registros (RecordSet) dentro de la base de datos:

Ejemplo: `Obj_RSusuario.Open "Usuario", Obj_Conn, 3, 3`

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

11

## Características ASP.

- La propiedad *Filter* se utiliza para seleccionar un conjunto de registros de acuerdo al valor de uno o más de sus campos:

Ejemplo: `Obj_RSusuario.Filter="email=' "&Email;" ' "`

- La propiedad *EOF* devuelve TRUE cuando se ha sobrepasado el último registro del RecordSet.

Ejemplo:

```
IF NOT(Obj_RSusuario.EOF) THEN
Clave2=Obj_RSusuario ("clave")
END IF
```

04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

12

# Características ASP.

- El método *AddNew* añade un nuevo registro al RecordSet.
- El método *Update* actualiza los cambios hechos en el RecordSet, ya sea por añadir un registro o modificar uno existente.

Ejemplo:

```
Obj_RS.AddNew  
Obj_RS ("nombre") = Nombre  
Obj_RS ("apellido") = Apellido  
Obj_RS ("edad") = Edad  
Obj_RS ("email") = Email  
Obj_RS ("pais") = Pais  
Obj_RS ("clave") = Clave1  
Obj_RS.Update
```

- El método *Close* cierra el RecordSet.

Ejemplo:

```
Obj_RSusuario.Close
```

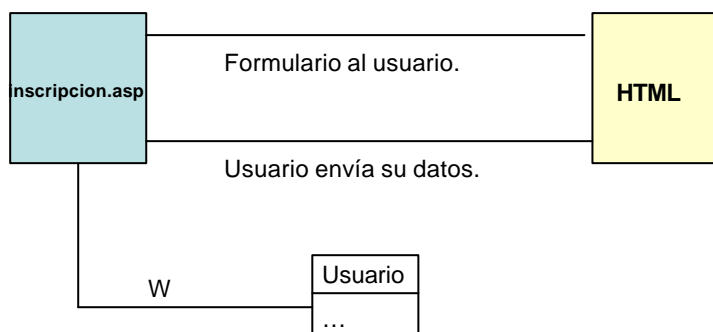
04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

13

# Implementación

- Página de inscripción.



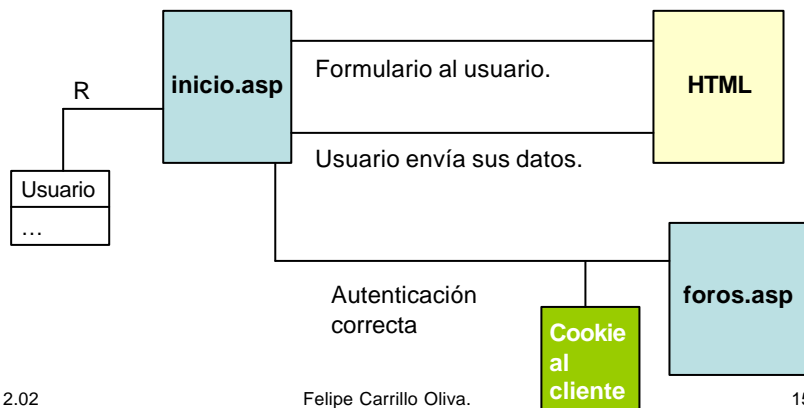
04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

14

# Implementación

- La página de inicio



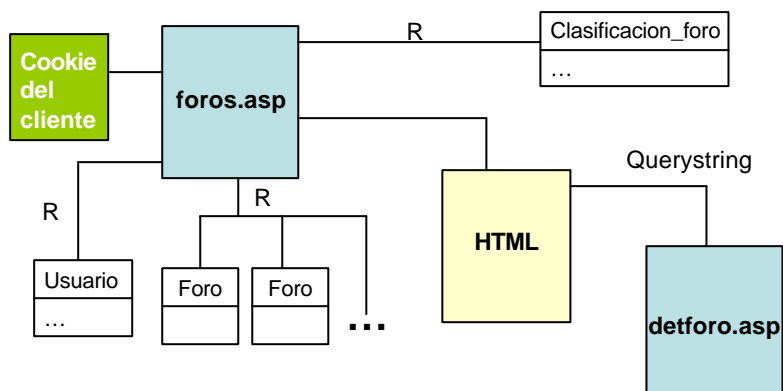
04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

15

# Implementación

- Inicio de sesión.



04.12.02

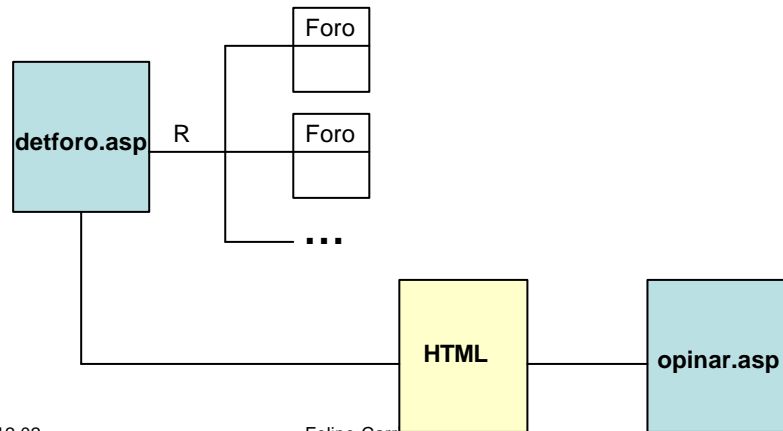
Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

16



# Implementación

- Pagina de detalle del foro.



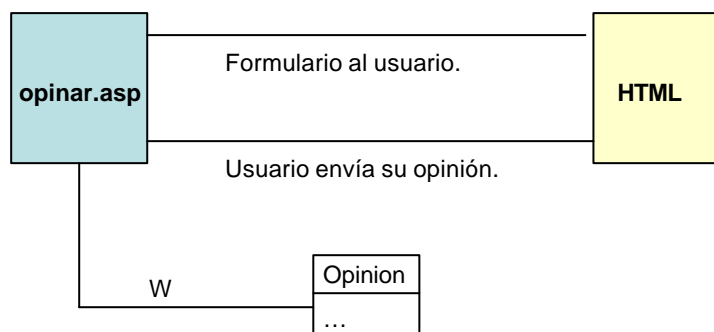
04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

17

# Implementación.

- Página para opinar.



04.12.02

Felipe Carrillo Oliva.  
ELO-330 Programación de Sistemas 2002.

18

# Conclusiones.

- Esta es una aplicación sencilla, y de diseño de principiante, pero muestra lo que se puede lograr con páginas web dinámicas y conexiones a bases de datos.

# Referencias

- **Active Server Pages (ASP 3.0) Iniciación y Referencia.**
  - Jesús Bobadilla Sancho. UPM.
  - Alejandro Alcocer Jarabe. NaI COM.
  - Luis Rodríguez-Manzaneque Sánchez. NaI COM.
  - Editorial McGraw Hill Interamericana. 2001.
- **Aprenda Visual Basic como si estuviera en primero.**
  - Javier García de Jalón.
  - José Ignacio Rodríguez.
  - Alonso Brazalez.
  - Escuela Superior de Ingenieros Industriales. Universidad de Navarra.1999.