

CAPÍTULO 4

DESARROLLO

4 Desarrollo

El proceso de desarrollo permite integrar los elementos de programación con los elementos gráficos con el fin de implementar un portal apegado a los objetivos y requerimientos planteados.

Es aquí donde se construye el portal, de tal manera que se apegue a las necesidades y expectativas del usuario.

El Portal de Administración y Control de Usuarios (PAyCU), inicialmente estará alojado en un servidor de prueba, los archivos y carpetas de la parte gráfica y de programación se encuentran almacenados en un equipo local. Una vez probado revisado corregido y vuelto a probar se procederá a usarlo de manera interna para algunos usuarios de la empresa, una vez pasado este período de pruebas se colocará el acceso en la página principal de Microbotix para su uso.

Dentro del desarrollo se consideraron los siguientes aspectos:

Disponibilidad. El sistema requiere estar disponible 24hrs al día los 365 días al año con una falla promedio de una hora al mes. Esto representa una disponibilidad de 99.86%

$$\frac{t * af}{t * af + tr} = \frac{24 * 30}{24 * 30 + 1} = \%99.86$$

Donde:

t * af: tiempo antes de la falla
tr: tiempo para reparar la falla

Internetwork nos da una garantía de disponibilidad de 99.9% por lo cual cumple con la disponibilidad requerida.

Modificabilidad. Microbotix es una compañía de formación reciente y cuyas necesidades van a cambiar conforme se de su crecimiento. Este crecimiento es esperable para el corto plazo. Dado lo anterior es absolutamente indispensable que el sistema sea fácilmente modificable.

Facilidad de uso. El usuario objetivo del sistema es de carácter no-técnico y por lo tanto la interfaz de usuario debe de ser muy fácil de manipular.

Por otro lado todas las funciones del sistema deben de estar a 4 o menos envíos de información por parte del usuario.

Facilidad de mantenimiento. Debido al ciclo iterativo incremental con el que va a ser desarrollado el sistema, es absolutamente necesario que el mantenimiento y corrección de errores sea fácil de llevar a cabo.

Escalabilidad. Dado que Microbotix es una empresa que espera tener un crecimiento constante en el corto plazo, el sistema debe de poder dar el mismo servicio a un creciente número de usuarios simultáneos, sin menores modificaciones que la ampliación de infraestructura de almacenamiento de bases de datos y agregación de módulos necesarios.

Portabilidad. Dado que el cliente del sistema utiliza un navegador de Internet la plataforma en la que se ejecute es irrelevante siempre que cumple con lo siguiente:

- Tenga un navegador compatible con los estándares de W3C
- El navegador debe de ser capaz de ejecutar código Javascript.

Creación Del Front-End y Back-End

Front-end es la parte de un sistema de software que interactúa directamente con el usuario; mientras que Back-end comprende los componentes que procesan la salida del Front-end. La separación de los sistemas de software en "front-end" y "back-end", es una abstracción que sirve para mantener separadas las diferentes partes de un sistema. (ALEGSA, 2011).

4.1 Implementación de la Base de Datos

Back-end

Una vez descargado e instalado EasyPHP nos creara un servidor para la base de datos con un manejador MySQL y un servidor Web apache para poder visualizar nuestra página.

La versión de EasyPHP para la implementación es 3.0 con:

- Apache 2.2.11
- PHP 5.2.8
- phpMyAdmin 3.1.1
- MySQL 5.1.30

Para proceder a la creación de la base de datos procedemos a ejecutar EasyPHP una vez ejecutado iniciará los servicios de Apache y MySQL, con clic derecho sobre el icono que se agrega a la barra de tareas activas en la parte inferior derecha, nos muestra una lista menú, vamos a dar clic en Administración.



Figura 40 Selecciona Administrador

Nos abrirá una página en un navegador con la siguiente dirección: <http://127.0.0.1/home/> la cual es la dirección local del sitio y a partir de la cual se podrá visualizar P AyCU

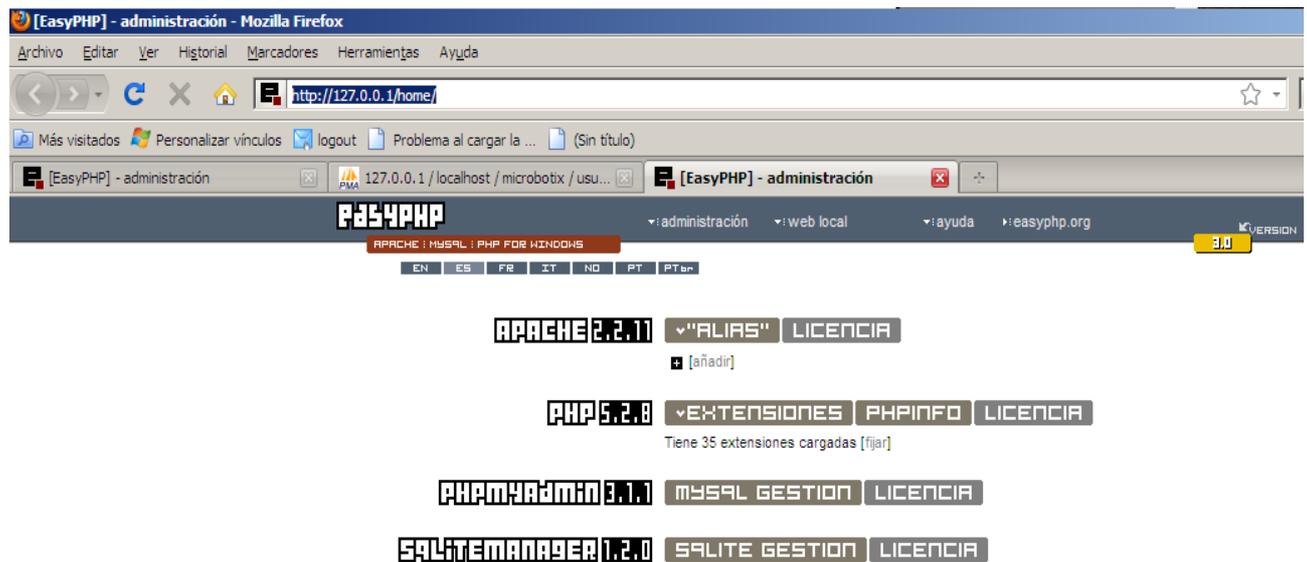


Figura 41 Interfaz de Administrador

Con clic en PHPmyAdmin accedemos a la interfaz gráfica que nos permite crear, modificar, eliminar y administrar de forma gráfica nuestra base de datos, sus tablas, relaciones, tipos de datos y valores. Se puede exportar la base, una vez creada, para poderla implementar en cualquier servidor, lo que nos permitirá migrar la base, cuando se instale en su sitio final. Podemos ejecutar consultas SQL de manera directa para probar los posibles resultado y podemos importar una base ya echa, si así lo necesitamos, entre otras cosas más.

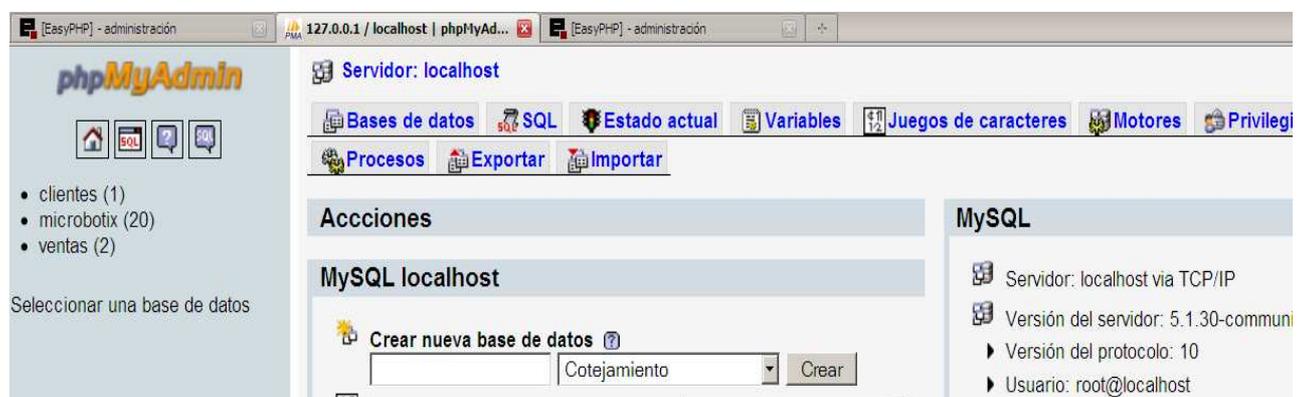


Figura 42 Interfaz Base de Datos

El nombre de la base de datos es *microbotix* y para la implementación del contenido que conforma la base de datos, se partió del diagrama Entidad Relación y del diccionario de datos, dando como resultado la creación de las tablas que muestra la figura:

127.0.0.1 / localhost / microbotix | phpMyAdmin 3.1.1 - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://127.0.0.1/home/mysq/index.php?db=microbotix&token=1d5169006f2ee8654526c5d32d509836

Más visitados Personalizar vínculos logout Problema al cargar la ... (Sin título)

127.0.0.1 / localhost / microbotix / microbotix... [EasyPHP] - administración

Servidor: localhost ▶ Base de datos: **microbotix**

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones

Privilegios Eliminar

Tabla	Acción			Registros ¹	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> administrador				2	InnoDB	utf8_general_ci	112.0 KB	-
<input type="checkbox"/> alumno				1	InnoDB	utf8_general_ci	32.0 KB	-
<input type="checkbox"/> asistencia_alumno				0	InnoDB	utf8_general_ci	32.0 KB	-
<input type="checkbox"/> asistencia_instructor				0	InnoDB	utf8_general_ci	32.0 KB	-
<input type="checkbox"/> bono				4	InnoDB	utf8_general_ci	16.0 KB	-
<input type="checkbox"/> cliente				0	InnoDB	utf8_general_ci	32.0 KB	-
<input type="checkbox"/> curso				0	InnoDB	utf8_general_ci	48.0 KB	-
<input type="checkbox"/> escuela				5	InnoDB	utf8_general_ci	48.0 KB	-
<input type="checkbox"/> grado_instructor				2	InnoDB	utf8_general_ci	16.0 KB	-
<input type="checkbox"/> grupo				0	InnoDB	utf8_general_ci	32.0 KB	-
<input type="checkbox"/> grupo_instructor				0	InnoDB	utf8_general_ci	64.0 KB	-
<input type="checkbox"/> grupo_usuario				6	InnoDB	utf8_general_ci	16.0 KB	-
<input type="checkbox"/> instructor				2	InnoDB	utf8_general_ci	96.0 KB	-
<input type="checkbox"/> matricula				0	InnoDB	utf8_general_ci	48.0 KB	-

Base de datos
microbotix (20)

- administrador
- alumno
- asistencia_alumno
- asistencia_instructor
- bono
- cliente
- curso
- escuela
- grado_instructor
- grupo_instructor
- grupo_usuario
- instructor
- matricula
- nivel
- puesto
- sucursal
- tipo_contrato
- tipo_pago
- usuario

Terminado

Figura 43 Vista Completa de la Interfaz de Base de datos con sus tablas

Se anexa el código para la creación de la base y las tablas más importantes, el cual se puede introducir directamente en la consola del administrador o hacerlo de forma gráfica.

Desde consola:

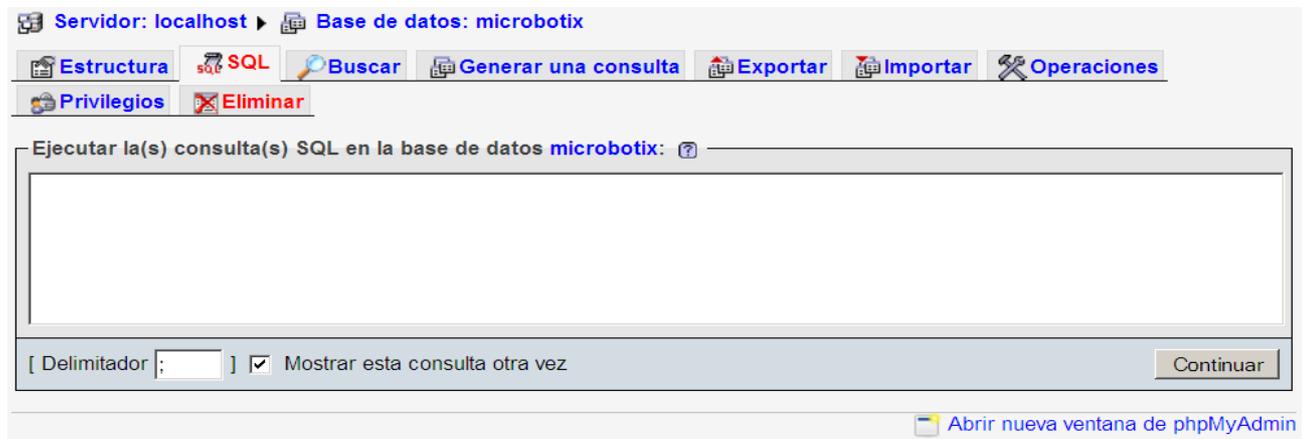


Figura 44 Ventana par introducir las Consultas a la base

Lenguaje SQL para la creación de tablas desde la base

Estructura de tabla para la tabla `instructor`

```
CREATE TABLE IF NOT EXISTS `instructor` (  
  `id_instructor` int(15) NOT NULL AUTO_INCREMENT,  
  `nombre_instructor` varchar(36) NOT NULL,  
  `ap_paterno` varchar(20) NOT NULL,  
  `ap_materno` varchar(20) DEFAULT NULL,  
  `edo_civil` varchar(15) DEFAULT NULL,  
  `carrera` varchar(30) DEFAULT NULL,  
  `fecha_ingreso` varchar(15) NOT NULL,  
  `fecha_nacimiento` varchar(15) NOT NULL,  
  `correo` varchar(30) NOT NULL,  
  `tel_casa` int(15) NOT NULL,  
  `tel_celular` int(15) DEFAULT NULL,  
  `rfc` varchar(13) NOT NULL,  
  `activo` int(1) NOT NULL,  
  `grado_escolar` varchar(15) NOT NULL,  
  `id_grado` int(3) NOT NULL,  
  `id_tipo_contrato` int(3) NOT NULL,  
  `id_tipo_pago` int(3) NOT NULL,  
  `id_usuario` int(15) NOT NULL,  
  PRIMARY KEY (`id_instructor`),  
  UNIQUE KEY `id_instructor` (`id_instructor`),  
  KEY `id_grado` (`id_grado`),  
  KEY `id_tipo_contrato` (`id_tipo_contrato`),  
  KEY `id_tipo_pago` (`id_tipo_pago`),  
  KEY `id_usuario` (`id_usuario`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;
```

Estructura de tabla para la tabla `administrador`

```
CREATE TABLE IF NOT EXISTS `administrador` (  
  `id_admin` int(15) NOT NULL AUTO_INCREMENT,  
  `nombre` varchar(36) NOT NULL,  
  `ap_paterno` varchar(20) NOT NULL,  
  `ap_materno` varchar(20) NOT NULL,  
  `edo_civil` varchar(15) DEFAULT NULL,  
  `carrera` varchar(30) DEFAULT NULL,  
  `fecha_ingreso` varchar(15) NOT NULL,  
  `fecha_nacimiento` varchar(15) NOT NULL,  
  `correo` varchar(30) NOT NULL,  
  `tel_casa` int(15) NOT NULL,  
  `tel_celular` int(15) DEFAULT NULL,  
  `tipo_administrador` int(3) NOT NULL,  
  `rfc` varchar(13) NOT NULL,  
  `id_puesto` int(3) NOT NULL,  
  `id_sucursal` int(3) NOT NULL,  
  `id_tipo_contrato` int(3) NOT NULL,  
  `id_tipo_pago` int(3) NOT NULL,  
  `id_usuario` int(15) NOT NULL,  
  `activo` int(1) NOT NULL,  
  PRIMARY KEY (`id_admin`),  
  UNIQUE KEY `rfc` (`rfc`),  
  KEY `id_puesto` (`id_puesto`),  
  KEY `id_sucursal`  
  (`id_sucursal`,`id_tipo_contrato`,`id_tipo_pago`,`id_usuario`),  
  KEY `id_tipo_pago` (`id_tipo_pago`),  
  KEY `id_usuario` (`id_usuario`),  
  KEY `id_tipo_contrato` (`id_tipo_contrato`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;
```

Estructura de tabla para la tabla `alumno`

```
CREATE TABLE IF NOT EXISTS `alumno` (  
  `id_alumno` int(15) NOT NULL AUTO_INCREMENT,  
  `nombre_alumno` varchar(36) NOT NULL,  
  `ap_paterno` varchar(20) NOT NULL,  
  `ap_materno` varchar(20) NOT NULL,  
  `fecha_nacimiento` varchar(15) NOT NULL,  
  `activo` int(1) NOT NULL,  
  `correo` varchar(30) NOT NULL,  
  `id_usuario` int(15) NOT NULL,  
  `tel_casa` int(15) NOT NULL,  
  `tel_celular` int(15) NOT NULL,  
  `grado_escolar` varchar(15) NOT NULL,  
  PRIMARY KEY (`id_alumno`),  
  KEY `id_usuario` (`id_usuario`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;
```

```

-- Estructura de tabla para la tabla `usuario`
--

CREATE TABLE IF NOT EXISTS `usuario` (
  `id_usuario` int(15) NOT NULL AUTO_INCREMENT,
  `login` varchar(40) NOT NULL,
  `password` varchar(40) NOT NULL,
  `id_grupo_usuario` int(15) NOT NULL,
  `foto` varchar(100) NOT NULL DEFAULT 'IMUsuario/robot2.jpg',
  PRIMARY KEY (`id_usuario`),
  UNIQUE KEY `login` (`login`),
  UNIQUE KEY `password` (`password`),
  UNIQUE KEY `login_2` (`login`,`password`),
  KEY `id_grupo_usuario` (`id_grupo_usuario`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=88 ;

ALTER TABLE `administrador`
  ADD CONSTRAINT `administrador_ibfk_10` FOREIGN KEY (`id_puesto`)
REFERENCES `puesto` (`id_puesto`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `administrador_ibfk_11` FOREIGN KEY (`id_sucursal`)
REFERENCES `sucursal` (`id_sucursal`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `administrador_ibfk_12` FOREIGN KEY (`id_tipo_contrato`)
REFERENCES `tipo_contrato` (`id_tipo_contrato`) ON DELETE CASCADE ON
UPDATE CASCADE,
  ADD CONSTRAINT `administrador_ibfk_13` FOREIGN KEY (`id_tipo_pago`)
REFERENCES `tipo_pago` (`id_tipo_pago`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `administrador_ibfk_14` FOREIGN KEY (`id_usuario`)
REFERENCES `usuario` (`id_usuario`) ON DELETE CASCADE ON UPDATE
CASCADE;
--
-- Filtros para la tabla `alumno`
--
ALTER TABLE `alumno`
  ADD CONSTRAINT `alumno_ibfk_4` FOREIGN KEY (`id_usuario`) REFERENCES
`usuario` (`id_usuario`) ON DELETE CASCADE ON UPDATE CASCADE;
--
-- Filtros para la tabla `asistencia_alumno`
--
ALTER TABLE `asistencia_alumno`
  ADD CONSTRAINT `asistencia_alumno_ibfk_3` FOREIGN KEY (`id_alumno`)
REFERENCES `alumno` (`id_usuario`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `asistencia_alumno_ibfk_4` FOREIGN KEY (`id_grupo`)
REFERENCES `grupo` (`id_grupo`) ON DELETE CASCADE ON UPDATE CASCADE;

```

```

-- Filtros para la tabla `asistencia_instructor`
--
ALTER TABLE `asistencia_instructor`
  ADD CONSTRAINT `asistencia_instructor_ibfk_3` FOREIGN KEY
(id_instructor`) REFERENCES `instructor` (id_instructor`) ON DELETE
CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `asistencia_instructor_ibfk_4` FOREIGN KEY (id_grupo`)
REFERENCES `grupo` (id_grupo`) ON DELETE CASCADE ON UPDATE CASCADE;
--
-- Filtros para la tabla `curso`
--
ALTER TABLE `curso`
  ADD CONSTRAINT `curso_ibfk_1` FOREIGN KEY (id_nivel`) REFERENCES
`nivel` (id_nivel`);

-- Filtros para la tabla `grupo`
--
ALTER TABLE `grupo`
  ADD CONSTRAINT `grupo_ibfk_1` FOREIGN KEY (id_escuela`) REFERENCES
`escuela` (id_escuela`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `grupo_instructor`
--
ALTER TABLE `grupo_instructor`
  ADD CONSTRAINT `grupo_instructor_ibfk_3` FOREIGN KEY (id_grupo`)
REFERENCES `grupo` (id_grupo`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `grupo_instructor_ibfk_4` FOREIGN KEY (id_instructor`)
REFERENCES `instructor` (id_instructor`) ON DELETE CASCADE ON UPDATE
CASCADE;

--
-- Filtros para la tabla `instructor`
--
ALTER TABLE `instructor`
  ADD CONSTRAINT `instructor_ibfk_5` FOREIGN KEY (id_grado`) REFERENCES
`grado_instructor` (id_grado`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `instructor_ibfk_6` FOREIGN KEY (id_tipo_contrato`)
REFERENCES `tipo_contrato` (id_tipo_contrato`) ON DELETE CASCADE ON
UPDATE CASCADE,
  ADD CONSTRAINT `instructor_ibfk_7` FOREIGN KEY (id_tipo_pago`)
REFERENCES `tipo_pago` (id_tipo_pago`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `instructor_ibfk_8` FOREIGN KEY (id_usuario`)
REFERENCES `usuario` (id_usuario`) ON DELETE CASCADE ON UPDATE
CASCADE;

--

```

```
-- Filtros para la tabla `matricula`
--
ALTER TABLE `matricula`
  ADD CONSTRAINT `matricula_ibfk_4` FOREIGN KEY (`id_alumno`)
REFERENCES `alumno` (`id_alumno`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `matricula_ibfk_5` FOREIGN KEY (`id_grupo`) REFERENCES
`grupo` (`id_grupo`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `matricula_ibfk_6` FOREIGN KEY (`id_cliente`)
REFERENCES `cliente` (`id_cliente`) ON DELETE CASCADE ON UPDATE
CASCADE;

--
-- Filtros para la tabla `usuario`
--
ALTER TABLE `usuario`
  ADD CONSTRAINT `usuario_ibfk_1` FOREIGN KEY (`id_grupo_usuario`)
REFERENCES `grupo_usuario` (`id_grupo_usuario`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

Desde el administrador en forma gráfica:

usuario 13 InnoDB utf8_general_ci

20 tabla(s) Número de filas 77 InnoDB utf8_general_ci

↑ Marcar todos/as / Desmarcar todos Para los elementos que están marcados: ▾

Vista de impresión Diccionario de datos

✱ Crear nueva tabla en la base de datos **microbotix**

Nombre: Número de campos:

Continuar

i 1 Podría ser aproximado. Léase la FAQ 3.11

Abrir nueva ventana de phpMyAdmin

Figura 45 campo para tabla

Aquí se agrega el tipo de datos, la longitud, si es nulo o no, alguna función, si es autoincrementable y comentarios entre otros atributos.

Servidor: localhost ▶ Base de datos: clientes ▶ Tabla: usuario

Campo	Tipo	Longitud/Valores ¹	Predeterminado ²	Cotejamiento	Atributos	Nulo	Índice	A_I	Comentarios
id_usuario	INT	15	Ninguna			<input type="checkbox"/>	--	<input type="checkbox"/>	
login	VARCHAR	40	Ninguna			<input type="checkbox"/>	--	<input type="checkbox"/>	
password	VARCHAR	40	Ninguna			<input type="checkbox"/>	--	<input type="checkbox"/>	
id_grupo_usuario	INT	15	Ninguna			<input type="checkbox"/>	--	<input type="checkbox"/>	
foto	VARCHAR	200	Ninguna			<input type="checkbox"/>	--	<input type="checkbox"/>	

Comentarios de la tabla:

Motor de almacenamiento: InnoDB

Cotejamiento:

definición de la PARTICIÓN:

Grabar O Añadir 1 campo(s)

Figura 46 Ingresar atributos

Ejemplo de creación de la tabla usuario:

Servidor: localhost ▶ Base de datos: clientes ▶ Tabla: usuario

Campo	Tipo	Longitud/Valores ¹	Predeterminado ²
id_usuario	INT	15	Ninguna
login	VARCHAR	40	Ninguna
password	VARCHAR	40	Ninguna
id_grupo_usuario	VARCHAR	15	Ninguna
foto	VARCHAR	200	Ninguna

Comentarios de la tabla: Ordenamiento: Cotejamiento:

NUMERIC
TINYINT
SMALLINT
MEDIUMINT
INT
BIGINT

Figura 47 Ejemplo creación de la tabla usuario

Front-end

Una vez construida la base de datos se procede a la construcción de la parte gráfica, que posteriormente será programada para interactuar con la base de datos. El diseño de la página se realizó en Macromedia Dreamweaver 8 que es un programa que nos permite la creación de páginas Web dinámicas y estáticas, así como editar códigos para diferentes lenguajes, otra herramienta la cual es gratuita y se trabaja desde un navegador es CKEditor, la cual, también nos permite el diseño de nuestra página Web de una forma gráfica, optimizando tiempos y resultados.

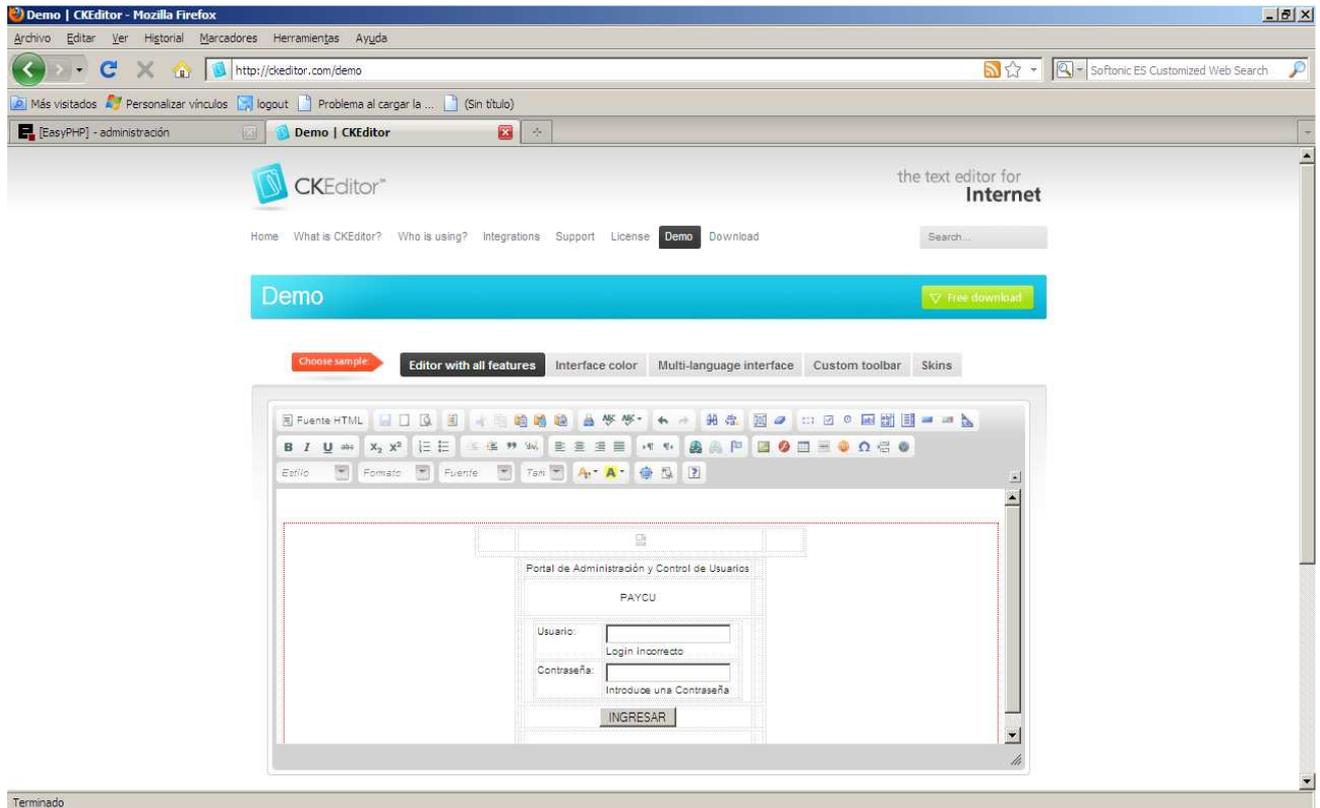


Figura 48 Página de CKEditor

Además de poder hacer el diseño de forma gráfica nos permiten visualizar el código de nuestra página para agregar funciones, incluir archivos o hacer correcciones directas sobre código, agregar algunas funciones de javascript e insertar estilos de una forma fácil y verlos al momento de agregarlos sin necesidad de subir la página al servidor Web. Además una vez terminado el sitio y echas las pruebas pertinentes tiene la opción de subir el sitio local a uno remoto a través de FTP.

Para visualizar las páginas en el navegador hay que guardar los archivos pertenecientes al sitio en la carpeta WWW dentro de EasyPHP donde fue creada después de la instalación. Se teclea la dirección 127.0.0.1 directamente en el navegado, o con clic derecho sobre el icono accedemos en el menú a Web local.

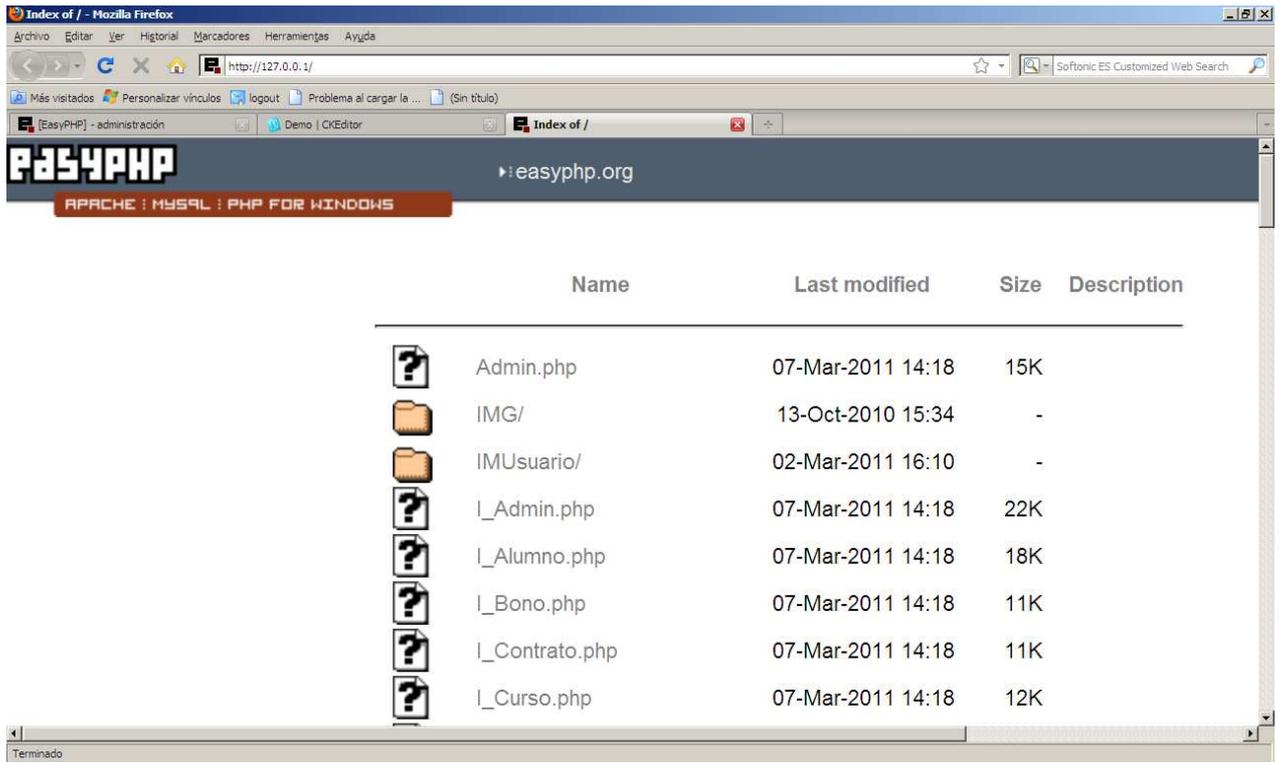


Figura 49 Servidor local

Pantalla de identificación para acceder al sistema:

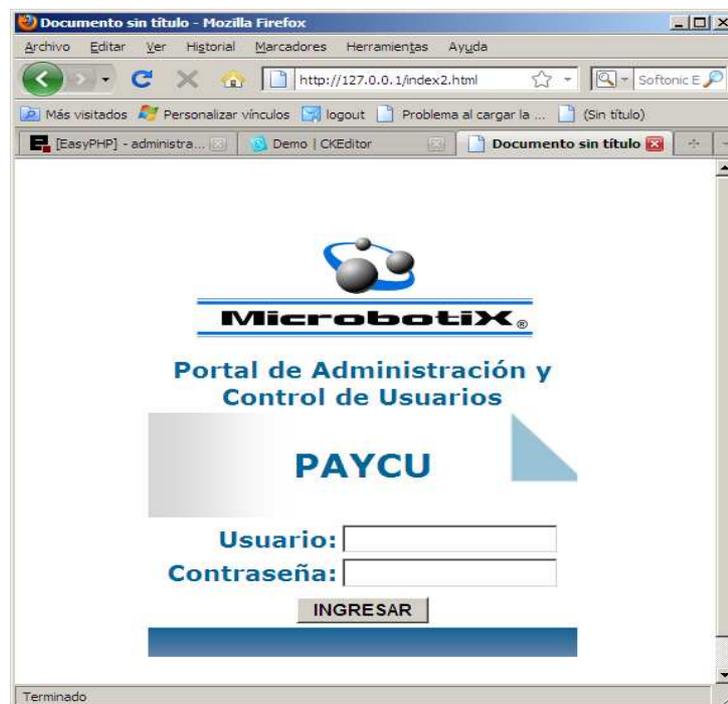


Figura 50 Pantalla inicio de Sesión

En esta pantalla es necesario que el usuario ingrese su nombre de usuario y contraseña para poder acceder al portal, en caso de no introducir la información correcta en alguno de los campos, desplegará un mensaje de error y no se tendrá acceso.

Para el portal, se diseñaron cuatro plantillas generales, una para cada perfil la cual se aplica a cada una de las páginas pertenecientes a este, cada una de estas plantillas parten de una plantilla general aplicable a todo el sitio. Las plantillas no son más que moldes o formatos que seguirán todas las páginas del sitio y son las que le darán forma a la interfaz de cada tipo de usuario.

La interfaz cambia de acuerdo al perfil del usuario, comparte y tiene diferentes opciones las cuales se muestran a continuación:

Administrador

Contiene un menú que le permite tener control total sobre el sistema, el puede crear modificar y eliminar a cada uno de los usuarios del sistema, bonos, niveles, sucursales, tipos de pago, tipos de instructor y grupos. Le permite ver los reportes de pagos asistencias deserciones y los generales, permite retener, liberar un pago y agregar una asistencia; además que podrá subir planeaciones y ver información de cualquier usuario. A continuación se muestran las pantallas más importantes.

The screenshot shows the instructor interface for a user named Silvia Nuñez Corona. The interface includes a navigation menu on the left, a user profile section, and a table of groups.

Reportes

- Pago
- Asistencia
- Deserción
- Generales

Ingresar

- Admin
- Instructor
- Escuela
- Alumno
- Grupo
- Nivel
- Bono
- Contrato
- Sucursal
- Puesto
- Tipo Pago
- Tipo

Modificar

- Admin
- Instructor
- Escuela
- Alumno
- Grupo
- Nivel
- Bono

Usuario: Administrador

Nombre: Silvia Nuñez Corona

Carrera: Ingeniería En Computación

Puesto: **Zona:**

Fecha Ingreso: 22/septiembre/2008

Fecha de nacimiento:

Correo:

Telefono: **Celular:**

Tipo de Administrador:

Grupo	Escuela	Horario	Nivel
24	Colegio Europeo	3:15 - 4:45	2
54	Colegio Europeo	3:15 - 4:45	2
10	Colegio Internacional	4:30 - 6:00	2

http://127.0.0.1/Ingresar_Tipo_Instructor.php

Figura 51 Interfaz del Instructor

Registro Administrador:

Cerrar Sesión

PAYCU Portal de Administración y Control de Usuarios

Bienvenido: silvia Usuario: Administrador 11-4-22 11 : 38 : 44

Registro Administrador

Datos Personales

* Apellido Paterno : * Apellido Materno: * Nombre(s):

Edo. Civil: Carrera: * Fecha de Nacimiento: ...

Elige una Opción * Telefono Casa: Telefono Celular: * Correo:

Datos del Empleado

* Puesto: * RFC: * Fecha de Ingreso: ...

Tipo de contrato: * Tipo de Pago :

Elige un Tipo Elige Uno

Tipo Administrador

* Tipo de Administrador: * Zona(s): * Activo: SI NO

Zona sur NORTE PONIENTE LOMAS LOMAS VERDES PONIENTE ESTE 3253654=){(}{(}{

Elige un Tipo

http://127.0.0.1/Ingesa_Tipo_Instructor.php

Figura 52 Registrar un Administrador

Ingresa Escuela:

INGRESA ESCUELA

Nombre Escuela :	<input type="text"/>
Dirección :	<input type="text"/>
Telefono 1 :	<input type="text"/>
Contacto 1 :	<input type="text"/>
Telefono 2 :	<input type="text"/>
Contacto 2 :	<input type="text"/>
Fax:	<input type="text"/>
Correo:	<input type="text"/>
Zona:	<input type="text" value="Zona"/>
Activo	Si <input type="radio"/> No <input type="radio"/>
Usuario	<input type="text"/>
Contraseña	<input type="text"/>
<input type="button" value="LIMPIAR"/> <input type="button" value="INGRESAR"/>	

Figura 53 Formulario para Ingresar Escuela

Ingresa Alumno:

Registro Alumno

Datos Personales

* Apellido Paterno :	* Apellido Materno:	* Nombre(s):																																																								
<input type="text" value="Molina"/>	<input type="text" value="Flores"/>	<input type="text" value="Leonardo"/>																																																								
Grado Escolar:	* Fecha de Nacimiento:	* Activo:																																																								
<input type="text" value="Prescolar"/>	<input type="text"/>	SI <input type="radio"/> NO <input checked="" type="radio"/>																																																								
* Telefono Casa:	Telefono Celular:	<table border="1"><thead><tr><th colspan="7">Abril, 2011</th></tr><tr><th colspan="7">Hoy</th></tr><tr><th>sem</th><th>Lun</th><th>Mar</th><th>Mié</th><th>Jue</th><th>Vie</th><th>Sáb Dom</th></tr></thead><tbody><tr><td>13</td><td></td><td></td><td></td><td></td><td>1</td><td>2 3</td></tr><tr><td>14</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9 10</td></tr><tr><td>15</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16 17</td></tr><tr><td>16</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23 24</td></tr><tr><td>17</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td></tr></tbody></table>	Abril, 2011							Hoy							sem	Lun	Mar	Mié	Jue	Vie	Sáb Dom	13					1	2 3	14	4	5	6	7	8	9 10	15	11	12	13	14	15	16 17	16	18	19	20	21	22	23 24	17	25	26	27	28	29	30
Abril, 2011																																																										
Hoy																																																										
sem	Lun		Mar	Mié	Jue	Vie	Sáb Dom																																																			
13					1	2 3																																																				
14	4	5	6	7	8	9 10																																																				
15	11	12	13	14	15	16 17																																																				
16	18	19	20	21	22	23 24																																																				
17	25	26	27	28	29	30																																																				
* Usuario:	* Contraseña:																																																									
<input type="text" value="leo"/>	<input type="text" value="....."/>																																																									
<input type="button" value="LIMPIAR"/> <input type="button" value="INGRESAR"/>																																																										

Figura 54 Formulario para Ingresar Alumno

Ingresar grupo:

Grupo Nuevo

#Grupo:

Fecha Inicio: ...

Fecha Fin: ...

Periodicidad: Lunes Martes Miercoles Jueves Viernes Sabado

Horario: - Hrs.

Escuela:

Asignación de Instructores :

Instructor SILVIA NUNEZ CORONA HGJK HGKLJ GFHKLJ	>>	
	<<	

#De Alumnos:

Grado Escolar:

Figura 55 Formulario Ingresar Grupo

Grado del instructor:

Grado Instructor

Tipo Instructor:

(Solo admite 1 Letra del Alfabeto)

Salario Por Hora :

Antigüedad: #

Promedio:

Vigente: Si No

LIMPIAR INGRESAR

Figura 56 Formulario Ingresa Instructor

Modifica administrador:

Busqueda Por :

Buscar

Buscar

Nombre login

Buscar

Figura 57 Modifica Instructor Búsqueda

Busco por Nombre:

Nombre(s): SILVIA NUNEZ CORONA

Limpiar Modificar

Figura 58 Modifica Instructor Selección

Una vez seleccionado, el sistema nos muestra los datos del usuario seleccionado a través de un formulario, para que modifiquemos lo que necesitemos y dejemos igual lo que no.

The screenshot displays a web form titled "Registro Administrador" with three main sections:

- Datos Personales:** Includes fields for "Apellido Paterno" (NUNEZ), "Apellido Materno" (CORONA), "Nombre(s)" (SILVIA), "Edad Civil" (Elige una Opción), "Carrera" (COMPUTO), "Fecha de Nacimiento" (15/03/2011), "Telefono Casa" (34534), "Telefono Celular" (4535), and "Correo" (sdf@nvc.ff).
- Datos del Empleado:** Includes fields for "Puesto" (gerente), "RFC" (NUCS121212RFT), "Fecha de Ingreso" (15/03/2011), "Tipo de contrato" (honorario), "Tipo de Pago" (HORAS), and "Declaración de Impuestos" (SI/NO).
- Tipo Administrador:** Includes fields for "Tipo de Administrador" (Administrador), "Zona(s)" (a list box with options like SUR, NORTE, PONIENTE, LOMAS, etc.), "Activo" (SI/NO), "Usuario" (slvirijilla), and "Contraseña" (masked).

At the bottom, there is a "PERFIL" icon and two buttons: "REESTABLECER" and "MODIFICAR".

Figura 59 Modifica Administrador

Una vez modificado damos clic en modificar y si todos los campos son correctos nos muestra la siguiente pantalla:



Figura 60 Aviso Ingreso Datos

Plantilla principal para Instructor

The screenshot displays the main interface for an instructor in the PAYCU system. At the top right, there is a green button labeled "Cerrar Sesion". Below this is a dark blue header bar with the "PAYCU" logo on the left and the text "Portal de Administración y Control de Usuarios" on the right. Underneath the header, four status indicators are shown: "Bienvenido: Silvia Núñez Corona", "Usuario: Instructor", "10 Octubre 2010", and "16:29".

A vertical navigation menu is located on the left side of the main content area. The menu items are: "Personal", "Pefil", "Pago", "Asistencia", "Deserción", "Evaluación", "Cursos", "Modelo Educativo", "Temario de Actividades", "Calendario", "Evidencias", "Alumnos", "Asistencia", "Evaluación", "Planeaciones", "Nivel 1", "Espacio", and "Fantasia". The "Pefil" item is currently selected and highlighted in green.

The main content area is a large, empty light gray rectangle, indicating that the specific content for the selected menu item is not yet loaded or is blank.

Figura 61 Interfaz Instructor

Plantilla principal para Escuela y Alumno:

PAYCU Portal de Administración y Control de Usuarios

Cerrar Sesión

Bienvenido: Colegio Internacional Usuario: Escuela 10 Octubre 2010 16:29

Asistencia
Evaluación
Modelo Educativo
Temario de Actividades
Nivel 1
Nivel 2
Nivel 3
Calendario
Ver Fotos
Pagos
Perfil de Instructores
Microbotix

Portal Para microbotix
Facultad de Ingeniería UNAM
Todos los derechos reservados

Silvia Núñez corona

<http://127.0.0.1/planeaciones/modeloeducativo.pdf>

Figura 62 Interfaz de Escuela y Alumno

Evaluaciones

Ingresa Evaluación:

FORMATO DE EVALUACIÓN DE ALUMNOS CURSOS ESCUELAS

Grupo	15
Nivel	2
Escuela	colegio internacional

	Nombre	Apellido Paterno	Apellido Materno	Concep	Prog	CT	Mtv	Intva	Comp	T.E
1	Jose Luis	Olivarez		1 ▾	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾
2	silvia	Núñez	Corona	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾
3	fernando	Mayorga	Juarez	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾	1 ▾

Conceptos	Concep
Programación	Prog
Circuitos	CT
Motivación	Mtv
Iniciativa	Intva
Comportamiento	Comp
Trabajo en Equipo	T.E

Figura 63 Ingresar Evaluación Alumno

Asuetos Académicos

Asuetos Académicos

Fecha
21/03/2011
05/02/2011

Agregar Día

Día 1 ▾ Mes Enero ▾ Año 2011 ▾

Figura 64 Formulario Ingreso Dias de Asueto

Muestra Evaluación:

EVALUACIÓN DE ALUMNOS

Evaluador:	Silvia Núñez Corona
Fecha Evaluación :	vptamos@yahoo.com.mx
Instructores del Curso:	Silvia y Rocío
Nivel	2
Grupo	15
Escuela	colegio internacional
Día	lunes
Horario	4:30-6:00

	Nombre	Apellido Paterno	Apellido Materno	C	P	CT	M	I	Comp	T.E
1	Jose Luis	Olivarez		3	3	2	2	2	3	3
Comentarios										
¡Felicidades JOSÉ LUIS, has llegado al final de tu primer curso de robótica! Hasta ahora has demostrado una capacidad sobresaliente para comprender conceptos relacionados con la robótica. En cuanto a tus habilidades para programar, nos gustaría felicitarte por tu extraordinario desempeño. Tu capacidad para armar circuitos eléctricos es realmente sorprendente. Nos gusta mucho ver que normalmente te muestras muy motivado por aprender cosas nuevas. Tu conducta es adecuada, pero deberías esforzarte un poco más para no distraerte con actividades que no van relacionadas a la clase. Vas muy bien en el curso en general y con un poco más de esfuerzo de tu parte podrás lograr resultados sobresalientes. ¡Esperamos verte de vuelta próximamente!										
2	silvia	Núñez	Corona	2	3	3	2	1	2	3
Comentarios										
¡Felicidades JOSÉ LUIS, has llegado al final de tu primer curso de robótica! Hasta ahora has demostrado una capacidad sobresaliente para comprender conceptos relacionados con la robótica. En cuanto a tus habilidades para programar, nos gustaría felicitarte por tu extraordinario desempeño. Tu capacidad para armar circuitos eléctricos es realmente sorprendente. Nos gusta mucho ver que normalmente te muestras muy motivado por aprender cosas nuevas. Tu conducta es adecuada, pero deberías esforzarte un poco más para no distraerte con actividades que no van relacionadas a la clase. Vas muy bien en el curso en general y con un poco más de esfuerzo de tu parte podrás lograr resultados sobresalientes. ¡Esperamos verte de vuelta próximamente!										

Figura 65 Evaluación de Alumnos Generada

Control Asistencia Instructor

Selecciona lista:

Cerrar Sesión

PAYCU Portal de Administración y Control de Usuarios

Bienvenido: Silvia Núñez Corona Usuario: Instructor 10 Octubre 2010 16:29

Consulta tu lista de asistencia

Grupo	Escuela	Horario	Nivel
24	Colegio Europeo	3:15 - 4:45	2
15	Colegio Internacional	4:30 - 6:00	2

Grupo: 24
Fecha: 07-Feb-09
Enviar

Importante:
No haz subido tu lista del grupo 24 de la semana 07-Febrero-09

Personal
Pefil
Pago
Asistencia
Deserción
Evaluación
Cursos
Modelo Educativo
Temario de Actividades
Calendario
Evidencias
Alumnos
Asistencia

Figura 66 Consulta Listas de Asistencia

Ingresando Asistencia:

Lista de asistencia

Grupo	Escuela			Horario	Nivel	
24	Colegio Europeo			3:15 - 4:45	2	
Nombre	Apellido Paterno	Apellido Materno	Fecha	Asistencia	Falta	Baja
1 Luis Daniel	Molina	Núñez	01/03/2009	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2 Fernado	Mayorga	Juarez	01/03/2009	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
3 Jesus Antonio	garcia	Hidalgo	01/03/2009	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Limpiar Subir

Figura 67 Ingresas Asistencia

Planeaciones:

The screenshot displays the PAYCU interface for lesson planning. The main content area is titled 'Insectos' and includes the following sections:

- Nivel:** 1 alta y baja
- Sesión:** 1
- Descripción:** Los niños entran al salón que tiene sonidos de naturaleza, mediante visualización los transportan a la jungla donde conocerán a los insectos que la habitan, ven los prototipos que van a hacer y se les enseña el comando de música, leds y pausa con la actividad "el escarabajo dice..." y el reto de programación en el hormiguero. Conocen las características de los insectos y hacen el hábitat de los insectos.
- Competencias:**
 - Resolución de problemas
 - Creatividad
 - Trabajo autónomo
 - Imaginación
 - Iniciativa
 - Trabajo en equipo
- Evidencia:**
 - Logro del reto
 - Preguntas de integración
 - Lluvia de ideas
- Conceptos:** Características y datos verídicos de los insectos, Conciencia en el cuidado de la naturaleza
- Programación:** Música, Leds
- Construcción:** (linked to Programación)
- Recursos:**
 - Decoración: lianas en el techo, plantas en las paredes, imágenes de insectos por todo el salón (robotix)
 - Música del modulo
 - Video IN THE JUNGLE
 - Video de los prototipos
 - Video, .ppt de insectos y datos verídicos
 - Material para el reto del hormiguero
 - .ppt "soy un insecto"
 - Batas, lentes (robotix)

A central image shows a robot with a spider-like body. A callout box points to it: "Los niños examinan el comando de música y leds por medio del logro del reto de la hormiga".

Figura 68 Planeación

Cursos:

The screenshot displays the 'Cursos' section of the PAYCU interface. The main content area is titled 'Nuestra motivación' and includes the following elements:

- Quote:** "El mundo en el que van a vivir nuestros hijos está cambiando cuatro veces mas rápido que nuestras escuelas" - Dr. William Daget [Director del International Center for Leadership and Education]
- Images:** Two photographs showing children in a classroom setting. The left photo shows two boys looking at a computer monitor. The right photo shows a group of children sitting around a table, engaged in an activity.
- Text:** buscamos preparar a los niños para vivir, crecer y trabajar en un mundo de constante evolución tecnológica.

Figura 69 Cursos

Temario de Actividades: para los alumnos y escuelas:



Figura 70 Temario de Actividades

Calendario de clases Microbotix:

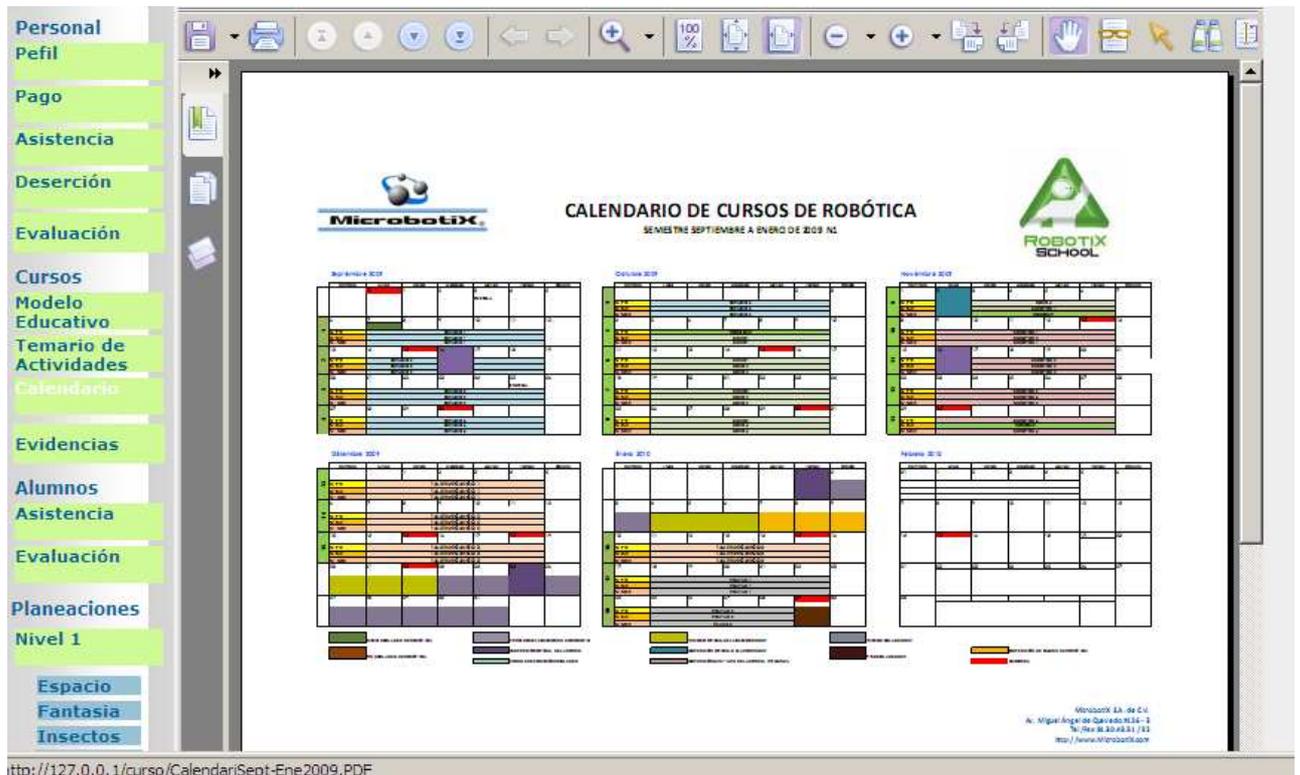


Figura 71 Calendario de Actividades

Reportes

El administrador selecciona el instructor, grupo y escuela, el sistema le muestra la información de pago perteneciente a ese instructor y le da la opción de modificar y agregar algún comentario antes de generar el reporte.

Para la generación de los reportes de pago, se accede a la siguiente interfaz para seleccionar el tipo de reporte a generar.

Figura 72 Seleccionar Tipo de Búsqueda

Figura 73 Selecciona una opción

Generar Reporte de Pago Por Instructor: En esta parte se ingresan bonos adicionales, horas de instrucción, horas de capacitación, comentarios y es aquí donde se libera o se retiene el pago de cada instructor de acuerdo a si ingreso su lisa a tiempo o no.

Reporte de Pago individual

Reporte generado:

Nombre:	Silvia Núñez Corona
Correo:	vptamos@yahoo.com.mx
Fecha:	22/septiembre/2008
Zona(s):	Norte, Sur
Apellido Paterno:	Daniel

Comentario

Consideraciones de Pago de Honorarios

Pago Por Hora de Clase	\$90
Pago por Hora de Capacitación	\$75
Pago por Bono de Transporte :	\$37.5
Régimen Fiscal del instructor :	HAS

Horas y días de Instrucción:

Fecha	Grupo	Asistencia	Horas Instrucción	Bono de Transporte	Responsable de Lista	Lista Ingresada	Liberar Pago
3/23/2009	24	1	1.5	1	SI	NO	NO
3/25/2009	15	1	1.5	1	NO	No Aplica	No Aplica
3/26/2009	10	0	0	0	SI	SI	No Aplica

Agrega Horas Instrucción: Ninguno, Motivo: []

Agrega Bono de Transporte: Ninguno, Motivo: []

Agregar Otro Tipo de Bono: Tipo: Ninguno, Horas: 0.5

Liberar Pago: Grupo: 14, Liberar: SI NO

Agregar Horas Capacitación: Fecha: 3/23/2009, Horas: 0.5

Limpiar Generar

Figura 74 Generar reporte de pago por instructor

Reporte de pago Instructor: Este reporte es el que puede consultar el instructor de manera informativa y el personal asignado cuando se desea verificar o consultar la información de manera oportuna para alguna aclaración.

Es un reporte detallado de las horas de instrucción, capacitación, número de bonos, su régimen fiscal y otros bonos por fecha. Al final se encuentra el monto total a cobrar para la fecha de corte seleccionada.

Reporte de Pago individual

Reporte generado:

Nombre:	Silvia Núñez Corona
Correo:	vptamos@yahoo.com.mx
Fecha Corte:	22/septiembre/2008
Zona(s):	Norte , Sur
Apellido Paterno:	Daniel

Comentario
 Esfuerzate mas llega más temprano y fomenta en tus niños ese gusto por aprender

Consideraciones de Pago de Honorarios	
Pago Por Hora de Clase	\$90
Pago por Hora de Capacitación	\$75
Pago por Bono de Transporte :	\$37.5
Régimen Fiscal del instructor :	HAS

Fecha	Horas Capacitación	Horas Instrucción	Bonos Transporte	Otro Bono	Importe Total Por Pagar
3/23/2009	1	0	0	0	\$1412.50
3/23/2009		1.5	1	0	
3/23/2009		1	1	1D	
Total		2.5	3	0	
Importe	\$75	\$225	\$112.50	\$1000	

Limpiar
Generar

Figura 75 Reporte de Pago Instructor generado

Reporte general de pago por escuela: En este reporte se obtiene el total de alumnos de todas las escuelas en las que imparte clases Microbotix, el total facturado de acuerdo al monto de la inscripción por el número de alumnos, más su colegiatura y el número de estados pagados por mes.

Nos muestra los datos de cobranza desglosados por mes para cada una de las escuelas.

Reporte General de Pago por Escuelas

Selección periodo Escolar: 2008-2009

Selección Mes: Septiembre

Limpiar Enviar

Total periodo Escolar 2008-2009 Septiembre					
Cobrar (Min)		Alumnos	Cantidad	Pagos	
Escuela	Grupo	Facturados	Facturada	Estado	
641	641	623	\$293,758.50	10	

SEPT 2008											
Mes 1		Listas		Alumnos		Pagos		Por Cobrar		Notas	
Nombre	Clave Escuela	Grupo	Inscripción	Precio Con iva	Escuela	Cobrar (Min)	Alumnos	Cantidad	Estado	Fecha de Pago	Cobrado
CA	ABERDEEN	42	----	452.00	1	10	37	\$26,874.60	1	09/10/08	\$26,874.60
		45	----	452.00	1	39					\$-----
		11	----	452.00	1	10					\$-----
		6	----	483.00	1	12					\$-----
CB	CHURCHILL	39	----	483.00	1	41	42	\$20,286.00	1	09/10/08	\$20,286.00
		9	----	483.00	1	7					\$-----
		54	----	483.00	1	11					\$-----
											NR129
											F1063

Regresar

Figura 76 Reporte general de pago por Escuela

Reporte Pago por Escuela: Es un reporte individual para cada una de las escuelas. Se elige el período escolar y el sistema despliega un reporte detallado, para cada mes, del número de alumnos facturados, la cantidad de facturación, la fecha de pago, lo que se ha cobrado, el número de factura y alguna nota importante para el área de contabilidad. Despliega también el total facturado que se lleva hasta el momento para el período seleccionado.

Reporte de Pago Por Escuela

Clave	Nombre Escuela	Grupo	Inscripción	Precio Con iva
CA	ABERDEEN	42	-----	452.00
		45	-----	452.00
		11	-----	452.00

Seleccione periodo Escolar: 2008-2009

Total periodo Escolar 2008-2009			
Cobrar (Min)		Cantidad	Pagos
Escuela	38	\$293,758.50	Estado 10

SEPT 2008										
Listas	Cobrar (Min)		Alumnos	Cantidad	Pagos		Fecha de Pago	Estado	Cobrado	Por cobrar
	Escuela	Grupo	Facturados	Facturada	Facturada	Pagos				
1		10								
1	39	19	37	\$26,874.60	1	1	09/10/08	\$26,874.60	NR129	
1		10								

Nota: Habrá un ajuste, pues finalmente se encontró que había 2 niños más. El pago se realizó en dos partes

OCT 2008										
Listas	Cobrar (Min)		Alumnos	Cantidad	Pagos		Fecha de Pago	Estado	Cobrado	Por cobrar
	Escuela	Grupo	Facturados	Facturada	Facturada	Pagos				
1		10								
1	39	19	37	\$26,874.60	1	1	09/10/08	\$26,874.60	NR129	
1		10								

Regresar

Figura 77 Reporte de Pago por Escuela

4.2 Programación segura de Páginas Web

Lo primero a considerar para el desarrollo de PAyCU fue la seguridad de cada una de las páginas así como mecanismos de autenticación, validaciones del lado del cliente, y validaciones del lado del servidor, este capítulo recibe este nombre debido a el contenido del portal, el cual trabajara con información confidencial de la empresa, información personal de los empleados e información de los clientes un manejo un tanto delicado. Es por eso que se definen los siguientes aspectos a considerar.

1. *Confidencialidad*. Que un usuario no pueda ver los datos de otro usuario que no le corresponden
2. *Integridad*. Que no puedan modificar los datos que no les correspondan ni el sistema

4.3 Integración de aplicación segura de acceso a los servicios

Debido a la naturaleza del sistema y a los diferentes tipos de usuarios, la información debe tener diferentes niveles de acceso:

- La información de carácter informativo debe ser accesible para el público en general
- La información relacionada con pagos, reportes, planeaciones, etc., debe ser accesible para los clientes y usuarios que posean ese perfil
- La información relacionada con la operación debe ser accesible solo para la empresa Microbotix
- La creación, modificación y eliminación debe ser permitida solo al administrador del sistema

Para la integridad, toda la información intercambiada entre clientes y servidor que contenga login y contraseña e información de pagos debe ser encriptada.

Algunas implementaciones de seguridad utilizadas:

Ocultar la información que se manda de página a página. Al seleccionar alguno de los grupos disponibles, para que nos lleve al detalle de sus alumnos, la problemática consiste en poder pasar de una página a otra el *id* del grupo seleccionado, sin que este pueda ser visible para el usuario final.

The screenshot shows a web browser window displaying a portal for 'PAYCU'. The page title is 'Reporte de Asistencia'. The interface includes a navigation sidebar on the left with categories like 'Reportes', 'Ingresar', and 'Modificar'. The main content area contains a table with the following data:

Nombre Escuela	Gpo	Nvl	Días	Inicio	Fin	Zona	Instructor Asignado	#A	\$I	\$P	%As
COLEGIO EUROPEO	2	2	MIERCOLES VIERNES	13:00	14:00	LOMAS VERDES	MAYORGA JUAREZ FERNANDO NUNEZ CORONA SILVIA	8	—	452	46.67
COLEGIO EUROPEO	3	2	MARTES JUEVES	16:00	17:00	LOMAS VERDES	MAYORGA JUAREZ FERNANDO NUNEZ CORONA SILVIA	4	—	452	50
COLEGIO IRLANDES	1	1	LUNES MIERCOLES	14:00	15:00	SUR	MARTINEZ LOPEZ GERVASIO NUNEZ CORONA SILVIA	6	345	500	19.44
COLEGIO IRLANDES	4	1	MIERCOLES VIERNES	14:00	15:00	SUR	MAYORGA JUAREZ FERNANDO	15	345	500	89.23

Below the table, there is a summary table with the following data:

Grupo	Gpo
Nivel	Nvl
Numero de alumnos	#A
Monto de Inscripción	\$I
Pago en Unidades Monetarias	\$P
Porcentaje de asistencia	%As

Figura 78 Página que envía información a otra

La primera solución para poder pasar la información de esta página a la otra, es pasar mediante el método GET, dentro del *href* de la etiqueta `<a>`, el valor del *id* del grupo, pero esto implicaría pasar mediante la URL tal valor, sería nuestro principal problema, ya que se muestra en la dirección el valor.

```
<a href="AsisAlumnoSeman.php?gp=<?=$reg['id_grupo']?>" target="_parent" class="Estilo11" >
<?=$reg['clave_grupo']?></a>
```

Para resolverlo, debemos de utilizar como método de envío el método POST. Y para poder utilizarlo se implemento una función de javascript que nos permite enviar, mediante un formulario oculto, el valor del *id* del grupo seleccionado.

```
<script>
function enviaval(pg){
    document.getElementById("pg").value = pg;
    document.frmenvia.submit();
}
</script>

<form action="AsisAlumnoSeman.php" method="post" name="frmenvia" target="_parent">
    <input type="hidden" name="pg" id="pg">
</form>
```

Validar formularios del lado del navegador. En esta tipo de validación se realizan las validaciones antes de que la información sea mandada al servidor, para esto se hizo uso de jQuery.

Figura 79 Validación de Formularios

Basta con incluir el archivo de la clase y mandarlo a llamar cada vez que lo necesitemos con la siguiente línea de código:

```
<script src="jquery-validate/lib/jquery.js" type="text/javascript"></script>
<script type="text/javascript" src="jquery-validate/jquery.validate.js"></script>

<!--<script type="text/javascript" src="vanadium_es.js"></script-->
<link href="screen.css" rel="stylesheet" type="text/css" />
```

Los archivos se almacenan en una carpeta en la raíz del sitio.

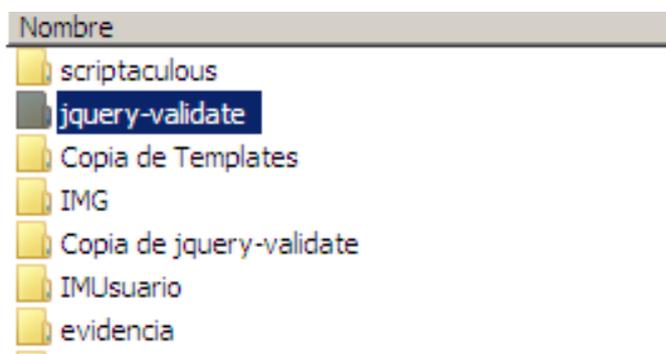


Figura 80 Ubicación de JQuery

La carpeta contiene una serie de archivos, cada uno de ellos tiene documentos con código editable el cual podemos ajustar a nuestras necesidades, en este caso se editó para el número telefónico y el RFC.

```
<!--script para el numero telefonico-->
<script>
$(document).ready(function(){
$.validator.addMethod("Numeros", function(value, element) {
return this.optional(element) || /^[0-9]+\$/i.test(value);
}, "Phone must contain only numbers, + and -.");

$("#form1").validate();
});
</script>
<!--fin script para el numero telefonico-->

<!--script para el USUARIO-->
<script>
$(document).ready(function(){
$.validator.addMethod("username", function(value, element) {
return this.optional(element) || /^[a-z0-9_]+\$/i.test(value);
}, "Usuario solo puede contener letras,numeros o underscore.");

$("#form1").validate();
});
</script>
<!--fin script para usuario-->

<!--script para Solo Letras-->
<script>
$(document).ready(function(){
$.validator.addMethod("letras", function(value, element) {
return this.optional(element) || /^[a-z]+\$/i.test(value);
}, "Usuario solo puede contener letras,numeros o underscore.");

$("#form1").validate();
}); </script>
<!--fin script Soloo letras-->

<!--script para RFC-->
<script>
$(document).ready(function(){
$.validator.addMethod("rfc", function(value, element) {
return this.optional(element) || /^[a-z]{4}([0-9]{6})([a-z0-9]{3})\$/i.test(value);
}, "las 4 primeras son letras los siguientes 6 numeros y los ultimos 3
alfanuméricos");

$("#form1").validate();
});
</script>
<!--fin script RFC-->
```

La ventaja es que se realiza directamente sobre el navegador, sin tener que ir al servidor, validar y esperar una respuesta, optimiza tiempo y se carga en la misma página.

Validaciones del lado del servidor. Para esta parte se realizaron una serie de funciones que se incluyeron en un archivo externo y en el mismo código de cada documento. Por ejemplo para eliminar caracteres raros o inyección de código malicioso en los campos de texto se creo la función limpia cadena.

```
<?php
function limpiacadena($cadena)

{
    $cadena = strip_tags($cadena);
    $cadena = str_replace ("*", "", $cadena);
    $cadena = str_replace ("% ", "", $cadena);
    $cadena = str_replace ("'", "", $cadena);
    $cadena = str_replace ("#", "", $cadena);
    $cadena = str_replace ("\\", "", $cadena);
    $cadena = str_replace("mysql","", $cadena);
    $cadena = str_replace("mssql","", $cadena);
    $cadena = str_replace("query","", $cadena);
    $cadena = str_replace("insert","", $cadena);
    $cadena = str_replace("into","", $cadena);
    $cadena = str_replace("update","", $cadena);
    $cadena = str_replace("delete","", $cadena);
    $cadena = str_replace("select","", $cadena);
    $cadena = str_replace("Character","", $cadena);
    $cadena = str_replace("MEMB_INFO","", $cadena);
    $cadena = str_replace("IN","", $cadena);
    $cadena = str_replace("OR","", $cadena);
    $cadena = str_replace (";", "", $cadena);
    //$cadena = str_replace ("", "", $cadena);

    return $cadena;
}
?>
```

Esta función se aplica de la siguiente manera:

```
$login=limpiacadena($_POST["login"]);//limpiando mi cadena de caracteres raros
$contra=(limpiacadena($_POST["contra"]));
if (!preg_match('/^[a-z\d_]{4,28}$/i', $login) and !preg_match('/^[a-z\d_]{4,28}$/i', $contra))
{
    echo "<P><font color='red'> Login o Contrase&ntilde;a invalidos</font> </P>";
}
}
```

Encriptar. Para encriptar la información que va a la base de datos se utilizo el algoritmo criptográfico MD5

La codificación del MD5 es de 128 bits y se representa como un número de 32 dígitos hexadecimal.

<pre>Echo \$Nombre="sil"; echo "
"; echo \$contra=md5('nunez'); echo "
";</pre>	<pre>d7f8f7ac0c50bb1cda75cc85e944d120</pre>
---	---

Aplicando la función md5 a contraseña

Contraseña encriptada