

Serv

```

1: <?PHP
2: //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
3: // scap/horfeus/index.php
4: // servidor scap con el servicio del nivel de registro de horfeus
5: //
6: // SERVICIO DE INFORMÁTICA Y COMUNICACIONES
7: // ÁREA DE APLICACIONES CORPORATIVAS
8: //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
9:
10: /* PARÁMETROS DE ENTRADA
11:    $solicitud    <horfeus>
12:                  <aulas><IP>xxx.xxx.xxx.xxx</IP></aulas>
13:                  <firma>
14:                    <IP>xxx.xxx.xxx.xxx</IP>
15:                    <TipoCliente>1|2|3|4</TipoCliente>
16:                    <tiempo>dd/mm/aaaa hh:mm:ss</tiempo>
17:                    <edificio>nn</edificio>
18:                    <TipoAula>nn</TipoAula>
19:                    <aula>xxxx</aula>
20:                    <usuario>UVUS</usuario>
21:                    <clave>****</clave>
22:                  </firma>
23:                </horfeus>
24:
25:    COMO RESPUESTA
26:    añade un hijo al xml de entrada con la estructura:
27:    <resultado>
28:      <dispositivo>nn</dispositivo>
29:      <lista>
30:        <aula>
31:          <codigo>xxxx</codigo>
32:          <edificio>nn</edificio>
33:          <tipc>n</tipc>
34:          <nombre>xxxxxxxxxxxx</nombre>
35:        </aula>
36:      </lista>
37:
38:      <registro>nnnn</registro>
39:      <tiempoServidor>dd/mm/aaaa hh:mm:ss</tiempoServidor>
40:      <documento>nnnnnnnn</documento>
41:      <nombre>xxxxxxxxxx</nombre>
42:    </resultado>
43: */
44:
45: /* HISTORICO
46:    05-09-2012 - demi          - creación del programa
47: */
48:
49: include('./xml.inc.php');
50:
51:
52: $xml = new miXml($_REQUEST['solicitud']);
53:
54: if($xml->error)
55: {
56:   $r = new miXml('<horfeus><error>XML mal formado</error></horfeus>');
57:   $r->salida();
58:   exit;
59: }
60:
61: if($xml->nombre!='horfeus')
62: {
63:   $r = new miXml('<horfeus><error>Etiqueta mal formada</error></horfeus>');
64:   $r->salida();

```

```
128:         case 'IP': $IP=$h->texto; break;
129:         case 'TipoCliente': $tipocliente=$h->texto; break;
130:         case 'tiempo': $tiempo=$h->texto; break;
131:         case 'edificio': $edificio=$h->texto; break;
132:         case 'TipoAula': $tipo=$h->texto; break;
133:         case 'aula': $aula=$h->texto; break;
134:         case 'usuario': $usuario=$h->texto; break;
135:         case 'clave': $clave=$h->texto; break;
136:         default: return errorfirma($peticion,$conexion,'Solicitud mal formada');
137:     }
138: }
139: if(empty($IP) ||
140:     empty($tipocliente) ||
141:     empty($tiempo) ||
142:     empty($edificio) ||
143:     empty($tipo) ||
144:     empty($aula) ||
145:     empty($usuario) ||
146:     empty($clave))
147:     return errorfirma($peticion,$conexion,'Faltan parámetros necesarios');
148:
149: list($correcto,$dispositivo,$aulas) = comprobarCliente($IP);
150: if(!$correcto)
151:     return errorfirma($peticion,$conexion,$aulas);
152:
153: $x=false;
154: if(is_array($aulas)) foreach($aulas as $a)
155:     if($edificio==$a[1] && $tipo==$a[2] && $aula==$a[0]) $x=true;
156: if(!$x)
157:     return errorfirma($peticion,$conexion,'Aula incorrecta');
158:
159: $doc = autenticaLDAP($usuario,$clave);
160: if(!$doc)
161:     return errorfirma($peticion,$conexion,'Identificación incorrecta');
162:
163: $sent = OciParse($conexion,
164:     "select decode(sexprs,'D','Dña.','D.') tratamiento,
165:         nomprs,lllprs,ll2prs
166:     from   tuib_persona,tuib_professor
167:     where  dniprs = '$doc'
168:     and    prs_dniprs = dniprs");
169: if(!OciExecute($sent))
170:     return errorfirma($peticion,$conexion,'Error al obtener profesor');
171: $n=OciFetchStatement($sent,$t);
172: if($n==0)
173:     return errorfirma($peticion,$conexion,'No encontrado como profesor');
174: $nombre = "{ $t['TRATAMIENTO'][0] } { $t['NOMPRS'][0] } { $t['LL1PRS'][0] }
175: { $t['LL2PRS'][0] }";
176:
177: $registro = null;
178: $fecha = null;
179: $sent = OciParse($conexion,
180:     "insert into tuse_cpp_registro
181:     (id_registro, fecha_origen, fecha_servidor, dis_id, aul_id,
182:     id_persona, prs_dniprs, tip_id, edi_codnum, tpl_codnum) values
183:     ( seq_tuse_cpp_registro.nextval,
184:     to_date('\".for_cadora($tiempo).\", 'dd/mm/yyyy hh24:mi:ss'),
185:     sysdate,$dispositivo,$aula,\".for_cadora($usuario).\",
186:     '$doc',$tipocliente,$edificio,$tipo'
187:     )
188:     returning id_registro,to_char(fecha_servidor,'dd/mm/yyyy
189: hh24:mi:ss') into :registro,:fecha");
190: OciBindByName($sent,':registro',&$registro,1000);
191: OciBindByName($sent,':fecha',&$fecha,1000);
```

```
254:         '127.0.0.1',
255:         '150.214.142.166',
256:         '150.214.143.116'
257:     ) as $p)
258:     if($_SERVER['REMOTE_ADDR']==$p) $x=true;
259: if(!$x)
260:     return array(false,null,'El proxy no está autorizado');
261:
262: if(!$conexion)
263:     return array(false,null,'Error en la conexión a los datos');
264:
265: $sent = OciParse($conexion,
266:     "SELECT D.DIS_ID,D.ESTADO
267:     FROM TUSE_CPP_DISPOSITIVO D
268:     WHERE D.DIRECCIONIP = '".for_cadora($ip)."'";
269: if(!OciExecute($sent))
270:     return array(false,null,'Error al obtener dispositivo');
271: $n=OciFetchStatement($sent,$t);
272: if($n==0)
273:     return array(false,null,'No se conoce el cliente '.$IP);
274: if($t['ESTADO'][0]!=1)
275:     return array(false,null,'El cliente no está activado');
276:
277: $dispositivo = $t['DIS_ID'][0];
278:
279: $sent = OciParse($conexion,
280:     "SELECT X.EDI_CODNUM edificio,
281:     X.TPL_CODNUM tipo,
282:     X.AUL_ID CODIGO,
283:     A.DESID2||' ('||E.DESID2||')' AULA
284:     FROM TUSE_CPP_DISPOSITIVO D,
285:     TUSE_CPP_CPSDISPAULA X,
286:     TUIB_AULA A,
287:     TUIB_EDIFICIO E
288:     WHERE D.DIS_ID = X.DIS_ID
289:     AND X.EDI_CODNUM = A.EDI_CODNUM
290:     AND X.TPL_CODNUM = A.TPL_CODNUM
291:     AND X.AUL_ID = A.ID_LUGAR
292:     AND E.CODNUM = A.EDI_CODNUM
293:     AND D.DIRECCIONIP = '".for_cadora($ip)."'";
294: if(!OciExecute($sent))
295:     return array(false,null,'Error al obtener aulas');
296: $n=OciFetchStatement($sent,$t);
297:
298: for($aulas=array(),$i=0;$i<$n;$i++)
299:     $aulas[] = array($t['CODIGO'][$i],$t['EDIFICIO'][$i],$t['TIPO'][$i],
300: $t['AULA'][$i]);
301:
302: return array(true,$dispositivo,$aulas);
303: }
304: function For_CadOra($cadena)
305: {
306:     return str_replace('\\"', '\"',
307:     str_replace('"', '\"',
308:     str_replace("\\\\", '\\',
309:     str_replace('\'', '\"', $cadena))));
310: }
311:
312: function autenticaLDAP($usuario,$clave)
313: {
314:     if($clave=='') return false;
315:
316:     if(($id=@ldap_connect('replica-ac.us.es'))==FALSE) return false;
```


cliente

```

1: <?PHP
2: //////////////////////////////////////
3: // soap/horfeus/cliente/index.php
4: // cliente soap con el servicio del nivel de captura de horfeus
5: //
6: // SERVICIO DE INFORMATICA Y COMUNICACIONES
7: // ÁREA DE APLICACIONES CORPORATIVAS
8: //////////////////////////////////////
9:
10: /* PARÁMETROS DE ENTRADA
11:     uvus_horfeus
12:     clave_horfeus
13:     aula_horfeus
14: */
15:
16: /* HISTORICO
17:     27-01-2014 - domi          - creación del programa
18: */
19:
20: include('./cabecera.php');
21: include('./xml.inc.php');
22:
23: $servidor = 'https://horfeus.us.es/registro/index.php';
24:
25: if(!empty($_POST['uvus_horfeus']) &&
26:     !empty($_POST['clave_horfeus']) &&
27:     !empty($_POST['aula_horfeus']) )
28: {
29:     list($e,$t,$a) = explode('+',$_POST['aula_horfeus'],3);
30:     $respuesta = file_get_contents("$servidor?solicitud=".
31:         urlencode(
32:             '<horfeus><firma>'.
33:             '<IP>'.$_SERVER['REMOTE_ADDR'].'</IP>'.
34:             '<TipoCliente>3</TipoCliente>'.
35:             '<tiempo>'.date('d/m/Y H:i:s').'</tiempo>'.
36:             '<edificio>'. $e .'</edificio>'.
37:             '<TipoAula>'. $t .'</TipoAula>'.
38:             '<aula>'. $a .'</aula>'.
39:             '<usuario>'. $_POST['uvus_horfeus'].'</usuario>'.
40:             '<clave>'. $_POST['clave_horfeus'].'</clave>'.
41:             '</firma>'.
42:
43:             '<aulas><IP>'.$_SERVER['REMOTE_ADDR'].'</IP></aulas></horfeus>'));
44:     $xml = new miXml($respuesta);
45:     if(!$xml)
46:     {
47:         errorfirma('No se ha obtenido respuesta');
48:         exit;
49:     }
50:     if($xml->error)
51:     {
52:         errorfirma($xml->error);
53:         exit;
54:     }
55:     $xmlAulas = null;
56:     $firma = null;
57:     if(is_array($xml->hijos)) foreach($xml->hijos as $hijo)
58:     {
59:         if($hijo->nombre=='aulas') $xmlAulas = $hijo;
60:         elseif($hijo->nombre=='firma') $firma = $hijo;
61:         else {errorfirma('Respuesta mal formada'); exit; }
62:     }
63:     if($firma)

```

```
127:     }
128:
129:     if($error)
130:     {
131:         error($error->texto);
132:         exit;
133:     }
134:
135:     if(!$respuesta)
136:     {
137:         error('No se ha obtenido respuesta');
138:         exit;
139:     }
140:
141:     $dispositivo=null;
142:     $lista=array();
143:     if(is_array($respuesta->hijos)) foreach($respuesta->hijos as $hijo)
144:     {
145:         if($hijo->nombre=='dispositivo') $dispositivo = $hijo->texto;
146:         if($hijo->nombre=='lista')
147:         {
148:             if(is_array($hijo->hijos)) foreach($hijo->hijos as $aula)
149:             {
150:                 if($aula->nombre=='aula')
151:                     $c=$n=$t=$e=null;
152:                 if(is_array($aula->hijos)) foreach($aula->hijos as $a)
153:                 {
154:                     if($a->nombre=='codigo') $c=$a->texto;
155:                     if($a->nombre=='edificio') $e=$a->texto;
156:                     if($a->nombre=='tipo') $t=$a->texto;
157:                     if($a->nombre=='nombre') $n=$a->texto;
158:                 }
159:                 $lista[]=array($c,$e,$t,$n);
160:             }
161:         }
162:     }
163:
164:     if(sizeof($lista)==0)
165:     {
166:         error('La lista de aulas está vacía<br>Confirme con el responsable del aula
que este equipo esté configurado');
167:         exit;
168:     }
169:
170:     ?>
171:     <form onsubmit="return chequear(this);" action="<?=$REQUEST_URI?>"
method="post" name="consulta">
172:         <div class="loginCont">
173:             <div class="filaLogin">
174:                 <label for="uvus">Aula</label>
175:                 <?php
176:                     if(sizeof($lista)==1)
177:                     {
178:                         ?>
179:                         <input type="hidden" name="aula_horfeus"
180:
value='<?=$lista[0][1]?>+<?=$lista[0][2]?>+<?=$lista[0][0]?>'>
181:                             <?=$lista[0][3]?>
182:                             <?php
183:                                 }
184:                             else
185:                             {
186:                                 ?>
187:                                 <select name="aula_horfeus"
onchange='document.consulta.uvus_horfeus.focus();'>
```

```
248:
249: include('./pie.php');
250:
251: function error($t)
252: {
253:     ?>
254:     <div id='ultima'>
255:         <h1>
256:             Error al obtener las aulas de este equipo<br>
257:             <?=$t?>
258:         </h1>
259:         <br><br><br><br><br><br><br><br>
260:     </div>
261: <?php
262: }
263:
264: function errorfirma($t)
265: {
266:     ?>
267:     <div id='ultima'>
268:         <h1>
269:             Error al realizar registro<br>
270:             <?=$t?>
271:         </h1>
272:         <br><br><br><br><br><br><br><br>
273:     </div>
274: <?php
275: }
276: ?>
```