

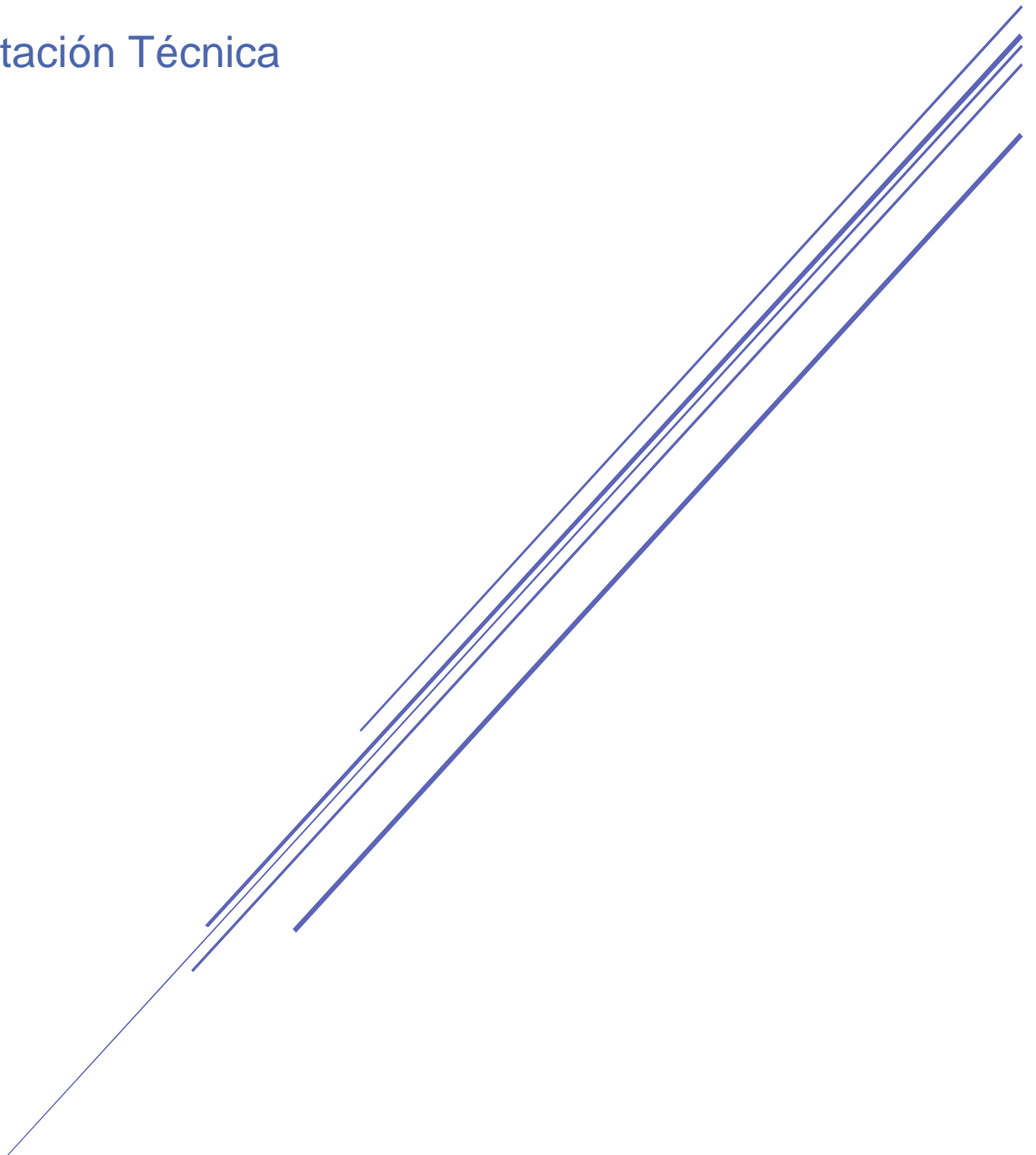
SCA

SISTEMA

CONTROL

ABSENTISMO

Documentación Técnica



Moisés Antonio Muñoz Aranda

Contenido

Acceso a la Aplicación2

Creación BD, tabla e inserciones de Prueba2

Principales ficheros, contenido y funcionalidad3

Acceso a la Aplicación

<http://moimuproyecto.ddns.net/>

<http://moimu.sytes.net/>

Para acceder a la parte privada: User/Pass jca/daw

Creación BD, tabla e inserciones de Prueba

La base de datos se llama controlabsentismo, y se compone de 4 tablas
su ubicación es pr101/00-bd/000_bd_tablas.sql

```
000_bd_tablas.sql X
codigodaw > pr101 > 00-bd > 000_bd_tablas.sql
1 CREATE DATABASE `controlabsentismo`;
2
3 CREATE TABLE `equipos` (
4     `idEquipo` MEDIUMINT UNSIGNED AUTO_INCREMENT,
5     `alias` VARCHAR(50) NOT NULL,
6     `descripcion` VARCHAR(50) ,
7     PRIMARY KEY (idEquipo)
8 );
9
10 CREATE TABLE `horarios` (
11     `idHorario` TINYINT UNSIGNED AUTO_INCREMENT,
12     `lunes` TINYINT,
13     `martes` TINYINT,
14     `miercoles` TINYINT,
15     `jueves` TINYINT,
16     `viernes` TINYINT,
```

También existe un juego de datos para pruebas
pr101/00-bd/fichero 001_inserciones.sql

```

001_inserciones.sql X
codigodaw > pr101 > 00-bd > 001_inserciones.sql
1  -- 7 equipos
2
3  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('Sin equipo','Para usuarios que no form
4  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('slash','Ventas');
5  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('gurruñico','Programadores');
6  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('pow','Atención cliente');
7  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('#','Diseño y Marketing');
8  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('@','Redes Sociales');
9  INSERT INTO `equipos`(`alias`,`descripcion`) VALUES ('wifi','Internacional');
10
11 -- 1 horario
12
13 INSERT INTO `horarios`(`lunes`,`martes`,`miercoles`,`jueves`,`viernes`) VALUES ('6','5','6',
14
15 -- 15 empleados
16
17 INSERT INTO `empleados`(`idEquipo`,`nombreCompleto`,`alias`,`email`) VALUES ('2','Moisés Ant
18 INSERT INTO `empleados`(`idEquipo`,`nombreCompleto`,`alias`,`email`) VALUES ('2','Golem Oper
19 INSERT INTO `empleados`(`idEquipo`,`nombreCompleto`,`alias`,`email`) VALUES ('3','Miguel Ang
20 INSERT INTO `empleados`(`idEquipo`,`nombreCompleto`,`alias`,`email`) VALUES ('3','Juan David

```

Principales ficheros, contenido y funcionalidad

Fichero: styles

- stylesindex.php
- stylesedicion.php
- stylesprivado.php

Fichero: apliPublic

- index.php

Contiene un menú de navegación común a todos los scripts, por el cual navegamos por las páginas de la aplicación.

```

    <link rel="icon" href="../images/favicon.ico">
</head>
<body>
    <!-- Encabezado con logo y menu navegador en la aplicación. -->
    <header class="header">
        <div class="containerlogomenu">
            <a href="#" class="logo"> S.C.A </a>
            <nav class="menu">
                <ol>
                    <li><a href="index.php"> Principal </a></li>
                    <li><a href="../apliPrivate/privado.php"> Privado Control Horario </a></li>
                    <li><a href="../apliPrivate/edicion.php"> Privado Edición </a></li>
                </ol>
            </nav>
        </div>
    </header>

```

Aquí se muestran datos públicos de los empleados en formato tabla
 Tiene asociada su hoja de estilos styles/stylesindex.php

```

<main>
    <div class="container">
        <?php
            /**
             * Muestro datos de usuarios activos con formato tabla para la parte pública de aplicación
             */
            include('../apliPrivate/conexionBd.php');

            $sentencia = $db->prepare("SELECT em.`idEmpleado`, em.`nombreCompleto`, em.`alias`, em.`email`, em.`
            FROM `empleados` em INNER JOIN `equipos` eq ON em.`idEquipo`= eq.`idEquipo` ");
            $sentencia -> execute();
            $sentencia -> bind_result($idEmpleado, $nombreCompleto, $aliasEmpleado, $email, $totalPuntos, $activo);

            echo "<table class=tabla1>";
            echo "<tbody><tr> <th>Alias</th> <th>Email</th> <th>Puntos de Mérito</th> <th>Equipo</th> </tr>";

            while($sentencia->fetch()){
                if($activo == 1){

```

Fichero: apliPrivate
 conexionBd.php

Mediante este script nos conectamos a la base de datos, por supuesto no olvidamos cerrar la conexión en nuestro código.

```

<?php
/**
 * conexion a la base de datos controlabsentismo de nuestra aplicación
 */
$servidor = "localhost";
$user = "root";
$password = null;
$database = "controlabsentismo";
$db = new mysqli($servidor,$user, $password,$database);
if($db->connect_error){
    die("La conexión con la bd ha fallado, error: " . $db->connect_errno . ": " . $db->connect_error);
}

```

Fichero: apliPrivate
edicion.php

Contiene formularios: para generar informes.

```
<form method="POST" enctype="application/x-www-form-urlencoded" action="edicion.php">
  <fieldset>
    <legend>Informes</legend>
    <div><label> Fecha: <input type="date" name="fecha" required></label></div>
    <div class="button"><button type="submit" name="enviar"> Generar </button></div>
  </fieldset>
</form>
```

Registro de nuevos empleados como usuarios a la aplicación

```
<form method="POST" enctype="application/x-www-form-urlencoded" action="edicion.php">
  <fieldset>
    <legend> Registro Nuevo Usuario</legend>
    <div><label> Nombre y apellidos: <input type="text" name="nombreCompleto" required></label></div>
    <div><label> Alias: <input type="text" name="alias" required></label></div>
    <div><label> Email: <input type="email" name="email" required></label></div>
    <div><label> Salario: <input type="number" min="0.00" max="99999.99" placeholder="3000.80" name="salario"></div>
    <div><label> Id Equipo: <input type="number" name="idequipo" min="1" placeholder=" 1 sin equi"></div>
    <div class="button"><button type="submit"> Generar </button></div>
  </fieldset>
</form>
```

Edición de usuarios existentes

```
<form method="POST" enctype="application/x-www-form-urlencoded" action="edicion.php">
  <fieldset>
    <legend> Editar Usuario</legend>
    <div><label> Id Empleado: <input type="number" name="idEmpleadoX" required></label></div>
    <div><label> Nombre y apellidos: <input type="text" name="nombreCompletoX" required></label></div>
    <div><label> Alias: <input type="text" name="aliasX" required></label></div>
    <div><label> Email: <input type="email" name="emailX" required></label></div>
    <div><label> Salario: <input type="number" min="0.00" max="99999.99" placeholder="3000.80" name="salario"></div>
    <div><label> Id Equipo: <input type="number" name="idequipoX" min="1" placeholder=" 1 sin equi"></div>
    <div class="button"><button type="submit"> Generar </button></div>
  </fieldset>
</form>
```

Alta o Baja de la aplicación, su funcionalidad principal es suspender a los empleados hasta la reincorporación a la empresa o simplemente, darlos de baja definitiva.

```

<form method="POST" enctype="application/x-www-form-urlencoded" action="edicion.php">
  <fieldset>
    <legend> Activar o Desactivar Usuario </legend>
    <div><label> Id Empleado: <input type="number" name="id" min="0" required><label></div>
    <div><label> Activar <input type="radio" name="activar" value="1" required><label></div>
    <div><label> Desactivar <input type="radio" name="activar" value="0" required><label></div>
    <div class="button"><button type="submit"> Generar </button></div>
  </fieldset>
</form>

```

Recogida fecha para generar informe, genero informo de nombre fichero y la ruta específica para acceder por navegador, además imprimo por pantalla con formato tabla para mayor comodidad del administrador.

```

if(isset($_POST['fecha'])){
    $sentencia = $db->prepare(" SELECT i.`fecha`, i.`totalPuntos`, e.`nombreCompleto`
    FROM `empleados` e INNER JOIN `informes` i ON e.`idEmpleado` = i.`idEmpleado` WHERE i.`fecha` = ? ");
    $sentencia -> bind_param('s', $param1);
    $param1 = $_POST['fecha'];
    $sentencia -> execute();
    $sentencia -> store_result();
    if($sentencia->num_rows > 0){
        $sentencia -> bind_result($fecha, $totalPuntos, $nombreCompleto);
        $fichero = fopen("informe_$param1.txt", "w");
        fwrite($fichero, "fechaInforme    totalPuntos    Nombre \n\n");
        echo "<table class=tableinforme border=1>";
        while($sentencia->fetch()){
            echo "<tr><td>$fecha</td><td>$totalPuntos</td><td>$nombreCompleto</td></tr>";
            fwrite($fichero, "$fecha          $totalPuntos          $nombreCompleto \n");
        }
        echo "</table>";
        $sentencia -> close();
        fclose($fichero);
        echo "<div class=smsImportante> Su informe se generó en directorio /apliPrivate/informe_$param1.txt </div>";
    }
    else{
        echo "<div class=smsImportante> No existe registro para fecha $param1, vuelva a intentarlo </div>";
    }
}

```

Registro de usuarios, el registro de usuarios se realiza correcta y eficazmente

```

if(isset($_POST['nombreCompleto'])){
    $sentencia = $db->prepare("INSERT INTO `empleados`(`idequipo`, `nombreCompleto`, `alias`, `email`, `salario`) V
    $sentencia -> bind_param('isssi', $param1, $param2, $param3, $param4, $param5);
    $param1 = $_POST['idequipo'];
    $param2 = $_POST['nombreCompleto'];
    $param3 = $_POST['alias'];
    $param4 = $_POST['email'];
    $param5 = $_POST['salario'];
    $sentencia -> execute();
    if($sentencia->affected_rows > 0){
        echo "<div class=smsImportante> Registro nuevo usuario Correcto </div>";
    }
    else{
        echo "<div class=smsImportante> No se registró ningún Usuario, vuelva a intentarlo </div>";
    }
    $sentencia -> close();
}

```

Edición de usuarios registrados, se realiza eficazmente.

```
if(isset($_POST['idEmpleadoX'])){\n    $sentencia = $db->prepare("UPDATE `empleados` SET `idequipo`=?, `nombreCompleto`=?, `alias`=?, `email`\n    $sentencia -> bind_param('isssii', $param1, $param2, $param3, $param4, $param5, $param6);\n    $param1 = $_POST['idequipoX'];\n    $param2 = $_POST['nombreCompletoX'];\n    $param3 = $_POST['aliasX'];\n    $param4 = $_POST['emailX'];\n    $param5 = $_POST['salarioX'];\n    $param6 = $_POST['idEmpleadoX'];\n    $sentencia -> execute();\n    if($sentencia->affected_rows > 0){\n        echo "<div class=smsImportante> Edicion Usuario Correcta <div>";\n    }\n    else{\n        echo "<div class=smsImportante> No se editó ningún Usuario, vuelva a intentarlo <div>";\n    }\n}
```

Actualización a baja o alta usuario de la aplicación, se realiza actualizando un campo `activo` en tabla `empleados`, con valores 0 o 1, por defecto los usuarios se generan activos = 1

```
if(isset($_POST['id'])){\n    $sentencia = $db->prepare("UPDATE `empleados` SET `activo` = ? WHERE `idEmpleado` =\n    $sentencia -> bind_param('ii',$param1, $param2);\n    $param2 = $_POST['id'];\n    $param1 = $_POST['activar'];\n    $sentencia -> execute();\n    if($sentencia->affected_rows > 0){\n        echo "<div class=smsImportante> Actualización Correcta <div>";\n    }\n    else{\n        echo "<div class=smsImportante> No se modificó el estado, ingrese id válido <div>";\n    }\n    $sentencia -> close();\n}
```

Fichero: apliPrivate
privado.php

Contiene formulario para la inserción de faltas y retrasos de forma conjunta a los empleados, realizado mediante datos consultados a tabla `empleados`.

```
$sentencia = $db->prepare("SELECT em.`idEmpleado`, em.`nombreCompleto`, em.`alias`, em.`email`, em.`totalPuntos`, em.\n    em.`activo`, eq.`alias` FROM `empleados` em INNER JOIN `equipos` eq ON em.`idEquipo`= eq.`idEquipo` ");\n    $sentencia -> execute();\n    $sentencia -> bind_result($idEmpleado, $nombreCompleto, $aliasEmpleado, $email, $totalPuntos, $prIncentivo, $salario);\n\n    echo "<table class=tabla1>";\n    echo "<form method=POST enctype=application/x-www-form-urlencoded action=privado.php >";\n\n    echo "<tbody><tr> <th>Activo</th><th>idEmpleado</th><th>Nombre</th><th>Alias</th> <th>Email</th><th>% Incentivo</th><th>\n    <th>Nº horas ausente</th><th>Nº horas retraso</th> </tr> ";
```



```

$idSeleccion = [];
$totalPuntosSeleccion = [];
$totalIds = 0;
while($sentencia->fetch()){

    echo "<tr><td>$activo</td><td>$idEmpleado</td><td>$nombreCompleto</td><td>$aliasEmpleado</td><td>$email</td><td>$salario</td><td>$totalPuntos</td><td>$aliasEquipo</td><td><input type=number min=0 name=ausente$totalIds required> </td> <td> <input type=number min=0 name=retraso$totalIds required> </td></tr>";
    $idSeleccion[$totalIds] = $idEmpleado;
    $totalPuntosSeleccion[$totalIds] = $totalPuntos;
    $totalIds++;
}
echo "</tbody>";
echo "</tfoot>";
echo "<tr class=trbutton> <td> Dia de la semana <input type=text name=dia_semana required> </td> <td><button type=submit> Guardar </button> </td></tr>";
echo "</tfoot>";
echo "</form>";
echo "</table>";
$sentencia->close();

```

Tratamiento de los datos consultados, ingresados en formulario, e inserción en tabla empleado e informes

Recojo valores en array de horas ausentes y retrasadas.

Calculo de total de horas ausentes + retrasos, y guardo en array
\$totalHausenteRetraso

```

for($i=0; $i<$totalIds; $i++){
    $ausente[$i] = intval($_POST['ausente'.$i]);
    $retraso[$i] = intval($_POST['retraso'.$i]);
    $totalHausenteRetraso [$i] = $_POST['ausente'.$i] + $_POST['retraso'.$i];
}
/**

```

Calculo de puntos totales a restar a cada empleado, multiplicando cada array de valores por su multiplicador correspondiente.

```

$multiplicadorHausente = 10;
$multiplicadorHretraso = 5;
for($i=0; $i<$totalIds; $i++){
    $puntosRestar[$i] = $ausente[$i] * $multiplicadorHausente + $retraso[$i] * $multiplicadorHretraso;
}
/**
 * Verificación de que $totalHausenteRetraso[] no contiene más de las horas laborales en el día en que

```

Consulta a tabla `horarios` y recogida de horas por día de semana, para este caso solo tenemos registro de horarios hábil la selección podría hacerla por empleados pertenecientes a un grupo concreto, por ejemplo, la selección para esta consulta es idEmpleado = 1, horario válido para todos los empleados.

```

$sentencia = $db->prepare("SELECT h.`lunes`, h.`martes`, h.`miercoles`, h.`jueves`, h.`viernes`
FROM `empleados` e INNER JOIN `horarios` h ON e.`idHorario` = h.`idHorario` WHERE `idEmpleado` = 1; ");
$sentencia->execute();
$sentencia -> bind_result($lun, $mar, $mie, $jue, $vie);
while($sentencia->fetch()){
    $horasDiaSemana = ["lunes" => $lun, "martes" => $mar, "miercoles" => $mie, "jueves" => $jue, "viernes" => $vie];
}
$sentencia->close();

```

Verificación de que \$totalHausenteRetraso[], no contiene más de las horas para día de la semana en que realizamos la actualización, si es así trigger = 0 y no actualizamos, mostramos error.

```

1
$diaSemana = $_POST['dia_semana'];
$trigger = 1;
foreach ($horasDiaSemana as $clave => $valor){
    if(strcasecmp($diaSemana,$clave) == 0){
        for($i=0;$i<$totalids;$i++){
            $puntosSumar[$i] = $valor - $totalHausenteRetraso[$i];
            if($totalHausenteRetraso[$i] > $valor){
                echo "<div class=smsImportante>NO SE ACTUALIZÓ Total horas introducidas
                Vuelva a introducirlas considerando las horas laborales contratadas
                $trigger = 0;
            }
        }
    }
}
/**

```

En caso contrario actualizamos, en caso que total puntos del empleado tratados sean menor a 50, actualizamos mediante nueva consulta que actualiza equipo a idEquipo = 1 Sin equipo y porcentajeIncentivo = 0 , En esta secuencia de actualización para todos los ids de los empleados, insertamos nuevo campo en tabla informes con los datos requeridos.

```
if($trigger == 1){
    $updatePuntosEmpleados = $db -> prepare("UPDATE `empleados` SET `totalPuntos` = ? WHERE `idEmpleado` = ?");
    $updatePuntosEmpleados -> bind_param('ii', $param1, $idEmpleado);

    $updatePuntosEquipoIncentivo = $db -> prepare("UPDATE `empleados` SET `totalPuntos`=?, `idEquipo`=?, `porcentajeIncentivo`=? WHERE `idEmpleado`=?");
    $updatePuntosEquipoIncentivo -> bind_param('iiii', $param1, $param2, $param3, $idEmpleado);

    $insertPuntosInformes = $db -> prepare("INSERT INTO `informes`(`idEmpleado`,`totalPuntos`,`fecha`) VALUES(?,?,?)");
    $insertPuntosInformes -> bind_param('iis', $idEmpleado, $param1, $param4);

    $param2 = 1;
    $param3 = 0;
    $param4 = date('Y-m-d');
```

```
for($i=0; $i<$totalids; $i++){
    $idEmpleado = $idSeleccion[$i];
    $param1 = $totalPuntosSeleccion[$i] - $puntosRestar[$i] + $puntosSumar[$i];
    if($param1 < 50){
        $updatePuntosEquipoIncentivo -> execute();
    }
    else{
        $updatePuntosEmpleados -> execute();
    }
    $insertPuntosInformes -> execute();
}

$updatePuntosEquipoIncentivo -> close();
$updatePuntosEmpleados->close();
$insertPuntosInformes -> execute();
echo "<div class=smsImportante> Puntos de Mérito de los usuarios actualizados </br>
Podrá generar un informe de este día </div>";
```