

Universidad Nacional de San Agustín

Facultad de Ingeniería de Producción y Servicios Escuela
Profesional de Ciencia de la Computación



Proyecto Final

ASIGNATURA: Trabajo Interdisciplinar

DOCENTE: Yessenia Deysi Yari Ramos

PERIODO ACADÉMICO: SEMESTRE 2022-A

GRUPO: A

INTEGRANTES: Eduardo German Ruiz Mamani

López Condori, Andrea del Rosario

Sergio Sebastian Santos Mena Quispe

Linghai Joaquin Zhong Callasi

AREQUIPA - PERÚ

2022

1. INTRODUCCIÓN

1.1 Presentación y objetivos

El objetivo del presente proyecto se enfocará en la implementación de una página web para registro de asistencias y calificaciones de los alumnos, para docentes de la Escuela Profesional de Ciencia de la Computación, que pueda facilitar el registro de las asistencias y calificaciones de los alumnos universitarios, también generar reportes por semana, mes o semestre, incluso calcular datos estadísticos, además poder disponer de la información en el momento que sea necesario.

1.2 Partes del Proyecto

ADMINISTRADOR

- Admin.php

PROFESOR

- Asistencia
- Tomar asistencia
- Informe
 - a. Informe
 - b. Gráficos
 - c. Pdf
- Calificaciones
- Editar Notas
- Ingresar Notas
- Portada
- Clase

ESTUDIANTE

- Clase
- Portada

LOGIN

- Login Portada (login.ph.)
- Validación del login (validar_login.php)
- JavaScript del Login (login.js)

ESTILOS

- Login
- Tablas
- Botones

2. AMINISTADOR

Admin.php

En la primera parte tenemos la inclusión de la base de datos, comenzamos con el encabezado, donde se muestra el título y los enlaces a los estilos y javascript. Siguiendo la parte del cuerpo que contiene el título de “Ingresar Clase”, se muestran cuatro elementos Html `<input>` . El primero contendrá el código del curso, el segundo, el semestre, el tercero el año y el cuarto nombre del curso. Pasamos a parte de “Turno o Grupo” aquí habrá tres elementos `<input type="checkbox">` con los respectivos grupos a,b y c (línea 22-28)

```
1  <?php include("base_datos.php"); ?>
2
3  <!DOCTYPE html>
4  <html lang="es">
5  <head>
6      <meta charset="UTF-8"/>
7      <title>Administrador</title>
8      <link rel="stylesheet" href="formulario_carpeta.css">
9      <script type="text/javascript" src="formulario_carpeta.js"></script>
10 </head>
11 <body>
12     <h2 style="text-align: center;">Ingresar Clase<br><br></h2>
13     <form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post" name="form1">
14         <input type="text" name="codigo" placeholder="Codigo del curso">
15         <br>
16         <input type="text" name="semestre" placeholder="Semestre">
17         <br>
18         <input type="text" name="year" placeholder="Año">
19         <br>
20         <input type="text" name="curso" placeholder="Nombre del curso">
21         <br>
22         <label>Turno o Grupo:</label>
23         <fieldset id="turnos" >
24             <legend>Turnos</legend>
25             <input id="A" name="turnos[]" type="checkbox" value="A"/>A<br>
26             <input id="B" name="turnos[]" type="checkbox" value="B"/>B<br>
27             <input id="C" name="turnos[]" type="checkbox" value="C"/>C<br>
28         </fieldset>
29         <br>
30         <input type="text" name="EP_1" placeholder="EP_1">
31         <br>
32         <input type="text" name="EP_2" placeholder="EP_2">
33         <br>
34         <input type="text" name="EP_3" placeholder="EP_3">
35         <br>
36         <input type="text" name="EC_1" placeholder="EC_1">
37         <br>
38         <input type="text" name="EC_2" placeholder="EC_2">
39         <br>
40         <input type="text" name="EC_3" placeholder="EC_3">
41         <br>
42         <input type="text" name="curso_1" placeholder="Llave al curso...">
43         <br>
44         <input type="text" name="curso_2" placeholder="Llave al curso...">
45         <br>
46         <input type="text" name="curso_3" placeholder="Llave al curso...">
47         <br>
```

Img.1

En la imagen 2 se creará una base de datos de este curso localizada en “ti_ciencias_computacion”, se conectará al base de datos con el método POST obtendremos lo que es el código, semestre, año, turnos, notas.

```

49         <button type="submit" id="btnAgregar">Crear</button>
50     </form>
51     <?php
52     $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
53     $BaseDatos->conectar();
54
55     if ($_SERVER["REQUEST_METHOD"] == "POST") {
56         $codigo = $_POST["codigo"];
57         $semestre = $_POST["semestre"];
58         $year = $_POST["year"];
59         $curso = $_POST["curso"];
60         $sep_1 = $_POST["EP_1"];
61         $sep_2 = $_POST["EP_2"];
62         $sep_3 = $_POST["EP_3"];
63         $sec_1 = $_POST["EC_1"];
64         $sec_2 = $_POST["EC_2"];
65         $sec_3 = $_POST["EC_3"];
66         $key_1 = $_POST["curso_1"];
67         $key_2 = $_POST["curso_2"];
68         $key_3 = $_POST["curso_3"];
69         $turnos = $_POST["turnos"];
70         for ($i = 0; $i < count($turnos); $i++) {
71             $BaseDatos->crearcursos($codigo, $turnos[$i], $semestre, $year, $curso, $sep_1, $sep_2, $sep_3, $sec_1, $sec_2, $sec_3, $key_1, $key_2, $key_3);
72         }
73     }
74
75     $BaseDatos->cerrar();
76     ?>
77
78 </body>
79 </html>

```

Img.2

3. PROFESOR

3.1 Asistencia

En la primera parte detenemos lo que es la asistencia, nos conectamos a la base de datos, obtenemos la clase en la que nos encontramos, luego los datos de los estudiantes, y la fecha del día actual, de la línea 23 a 37 es la aprte de insertar un nueva fila para asistencia

```

1  <?php include "Includes/Header.php" ?>
2
3  <body>
4
5
6  <h2 style="text-align: center;">Estudiantes Registrados<br><br></h2>
7  <button style="float: right;" onclick="location='Tomar_Asistencia.php?clase=<?php echo $_GET['clase'] ?>'">Tomar
Asistencia</button>
8  <table id="tablaUsuarios" class="tabla">
9      <?php
10
11      /* Conexion con la base de datos */
12      $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
13      $BaseDatos->conectar();
14
15      /* Se obtiene la clase en la que nos encontramos */
16      $clase = strtolower($_GET["clase"]);
17      /* Se obtiene lista de estudiantes */
18      $estudiantes = $BaseDatos->getEstudiantes($clase);
19      /* Se obtiene la fecha */
20      $Date = date('d_m_Y',time());
21
22      /* Se itera dentro de la lista de estudiantes */
23      if(!is_null($estudiantes)) {
24          if ($_SERVER["REQUEST_METHOD"] == "POST") {
25              $BaseDatos->insesion($clase);
26
27              while ($row = mysqli_fetch_assoc($estudiantes)) {
28                  $valor = $_POST[$row["cui"]];
29                  $cui = $row["cui"];
30                  /* Se inserta la nueva asistencia según la fecha correspondiente */
31                  $BaseDatos->insasistenciaclase($clase, $valor, $Date, $cui);
32              }
33
34              /* Se actualiza la lista de asistencia */
35              $estudiantes = $BaseDatos->getEstudiantes($clase);
36          }
37      }
38  }

```

Img. 1

Luego sigue la parte de realizar las tablas con el CUI, nombres, apellidos y las sesiones dadas mediante la función `getColumnasClases`. Después de eso va a aparecer todos los datos de los estudiantes, además de las asistencias y faltas que haya tenido, todo esto se muestra en la imagen 1.

```

41     echo "<thead>";
42     echo "<tr>";
43     echo "<th>CUI</th>";
44     echo "<th>Nombre</th>";
45     echo "<th>Apellidos</th>";
46
47     $sesiones = $BaseDatos->getColumnasClases($clase);
48     if(!is_null($sesiones)) {
49         while ($row = mysqli_fetch_assoc($sesiones)) {
50             if ($row['column_name'] != 'cui') {
51                 echo "<th>" . $row['column_name'] . "</th>";
52             }
53         }
54     }
55
56     echo "</tr>";
57     echo "</thead>";
58     echo "<tbody>";
59
60     if(!is_null($estudiantes)) {
61         while ($row = mysqli_fetch_assoc($estudiantes)) {
62             echo "<tr>";
63             echo "<td class='CUI'>" . $row["cui"] . "</td>";
64             echo "<td class='nombre'>" . $row["nombre"] . "</td>";
65             echo "<td class='apellido'>" . $row["apellido"] . "</td>";
66
67
68             $asisxest = $BaseDatos->getAsistEstudiantes($clase, $row["cui"]);
69             if(!is_null($asisxest)) {
70                 $row_ase = mysqli_fetch_assoc($asisxest);
71                 $sesiones = $BaseDatos->getColumnasClases($clase);
72                 while ($row_ses = mysqli_fetch_assoc($sesiones)) {
73                     if ($row_ses['column_name'] != 'cui') {
74                         echo "<td>".$row_ase[$row_ses['column_name']]."</td>";
75                     }
76                 }
77             }
78             echo "</tr>";
79         }
80     }
81
82     echo "</tbody>";
83
84     $BaseDatos->cerrar();
85     ?>
86 </table>
87 </body>
88 </html>

```

Img. 1

3.2 Tomar asistencia

En esta parte es donde se tomará la asistencia que comienza con dos buttons, el primero es para volver a la portada donde aparecen la tabla de los datos de los estudiantes. El segundo button es parte de un form, este tiene la funcionalidad de guardar toda la información de la tabla. Nos conectamos a la base de dato, extraemos la información del estudiante y la asistencias que ha tenido anteriormente.

```

1  <?php include "Includes/Header.php" ?>
2
3  <body>
4      <h2 style="text-align: center;">Estudiantes Registrados<br><br></h2>
5      <button style="margin: 0 auto;" onclick="location='Asistencia.php?clase=<?php echo $_GET['clase'] ?>'">Volver</button>
6
7      <form method="post" action="Asistencia.php?clase=<?php echo $_GET['clase'] ?>">
8          <button type="submit" id="btnTomarAssist">Guardar</button>
9          <table id="tablaUsuarios" class="tabla">
10             <?php
11                 /* Conexion con la base de datos */
12                 $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
13                 $BaseDatos->conectar();
14
15                 /* Se extrae la información de los estudiantes y sus asistencia */
16                 $clase = strtolower($_GET["clase"]);
17                 $estudiantes = $BaseDatos->getEstudiantes($clase);
18                 $sesiones = $BaseDatos->getColumnasClases($clase);
19                 $Date = date('d_m_Y',time());
20
21                 echo "<thead>";
22                 echo "<tr>";
23                 echo "<th>CUI</th>";
24                 echo "<th>Nombre</th>";
25                 echo "<th>Apellidos</th>";
26
27                 if(!is_null($sesiones)) {
28                     while ($row = mysqli_fetch_assoc($sesiones)) {
29                         if ($row['column_name'] != 'cui') {
30                             echo "<th>" . $row['column_name'] . "</th>";
31                         }
32                     }
33                 }
34             }
35

```

Img. 1

En la imagen dos se va a visualizar la contención de los datos de los estudiantes y sus asistencias, luego de la línea 60 a 72, nos va mostrar otra fila más donde el docente tendrá las opciones de poner F si es falta o P si está presente.

```

36         echo "<th>$Date</th>";
37
38         echo "</tr>";
39         echo "</thead>";
40         echo "<tbody>";
41
42         if(!is_null($estudiantes)) {
43             while ($row = mysqli_fetch_assoc($estudiantes)) {
44                 echo "<tr>";
45                 echo "<td class='CUI'>" . $row["cui"] . "</td>";
46                 echo "<td class='nombre'>" . $row["nombre"] . "</td>";
47                 echo "<td class='apellido'>" . $row["apellido"] . "</td>";
48
49                 $asisxest = $BaseDatos->getAsistEstudiantes($clase, $row["cui"]);
50                 if(!is_null($asisxest)) {
51                     $row_axe = mysqli_fetch_assoc($asisxest);
52                     $sesiones = $BaseDatos->getColumnasClases($clase);
53                     while ($row_ses = mysqli_fetch_assoc($sesiones)) {
54                         if ($row_ses['column_name'] != 'cui') {
55                             echo "<td>" . $row_axe[$row_ses['column_name']] . "</td>";
56                         }
57                     }
58                 }
59
60                 echo "<td>
61                     <select name='". $row["cui"] . "'>
62                     <option value=' ' > </option>
63                     <option value='P'>P</option>
64                     <option value='F'>F</option>
65                     </select>
66                     </td>";
67
68                 echo "</tr>";
69             }
70         }
71
72         echo "</tbody>";
73
74         $BaseDatos->cerrar();
75         ?>
76
77         </tbody>
78     </table>
79 </form>
80 </body>
81 </html>

```

Img. 2

3.3 Informe

d. Informe

En esta parte es el botón que nos llevará a la parte de informes, en la cual se va a obtener las tablas de asistencia, los cursos y las clasificaciones, primero se obtiene la clase y luego las demás, estos datos son enviados.

```

1  /* Boton que lleva al informe */
2  window.addEventListener("load",e => {
3      const btn = document.getElementById("btn-informe")
4      btn.addEventListener("click",e=>{
5          /* Se obtiene la clase */
6          let clase = btn.classList[0].slice(1,-1),
7
8          /* Se obtienen las tablas de asistencia, cursos y calificaciones */
9          tablaAsistencia = recibirDatos(clase,"asistencia"),
10         tablaCurso = recibirDatos(clase),
11         tablaCalificaciones = recibirDatos(clase,"calificaciones");
12         tablaDatos = recibirDatos(clase,"datos");
13         tablaEstadistica = recibirDatos(clase,"informacion_y_estadistica");
14
15         /* Se envian los datos obtenidos */
16         enviarDatos(tablaAsistencia,tablaCurso,tablaCalificaciones,tablaDatos,tablaEstadistica)
17     })
18 })
19
20

```

Img. 2

En la imagen 3 nos muestra la función de enviar datos, con los parámetros de tabla de Asistencias, Curso, Calificaciones, Datos y estadísticas) cada parámetro estará nombrado como param1, param2, param3, param4, para5 y param6, la localizacion sera el “reporte.php” + todos los parámetros.

```
23 async function enviarDatos(tablaAsistencia,tablaCurso,tablaCalificaciones,tablaDatos,tablaEstadistica){
24
25
26     let param1 = await getAbandonos(tablaAsistencia),
27     param2 = await getAsistenciaPorClase(tablaAsistencia),
28     param3 = await getNumeroClases(tablaCurso),
29     param4 = await getAprobados(tablaCalificaciones),
30     param5 = await getDatosCalificaciones(tablaCalificaciones,tablaDatos),
31     param6 = await getLimitesCalificaciones(tablaEstadistica);
32
33
34     window.location.href = "reporte.php" +
35     ">totalAsistencia=" + JSON.stringify(param1) +
36     "&asistenciaPorClase=" + JSON.stringify(param2) +
37     "&numeroClases=" + JSON.stringify(param3) +
38     "&calificacionesFinales=" + JSON.stringify(param4) +
39     "&datosCalificaciones=" + JSON.stringify(param5) +
40     "&limitesCalificaciones=" + JSON.stringify(param6);
41 }
42
```

Img. 3

En esta parte (imagen 4) aparece la función getAbandono que tiene como parámetro la tabla de la asistencia, en el cual todos los valores inicializan en 0, llega la parte del controlador, este almacenará el número de las clases asistidas por alumnos durante el semestre (línea 70 a 84).

```
57 async function getAbandonos(tablaAsistencia){
58
59     /* Se inicializan los valores en cero */
60     let array = [
61         {
62             label: "Abandonos",
63             y: 0
64         },{
65             label: "Asistentes",
66             y: 0
67         }
68     ]
69
70     tablaAsistencia.then(data => {
71         data.forEach(alumno => {
72             /*
73              Contador que almacenará el número de clases asistidas
74              en el semestre por alumno
75              */
76             let contadorAsistencias = 0;
77             for(const [key,value] of Object.entries(alumno)){
78                 if(key != "cui"){
79                     if(value == 'P') contadorAsistencias++;
80                 }
81             }
82             contadorAsistencias == 0 ? array[0].y++ : array[1].y++
83         })
84     })
85
86     return array;
87
88 }
```

Img.4

En esta imagen 5 esta la función AsistenciaPorClase, en el cual se van a obtener las fechas, seguido de un arreglo que guardará estos datos:

- Fecha
- Presentes
- Ausentes

De la línea 115 a 119 se va a evaluar si el alumno está presente o ausente.

```
127  async function getNumeroClases(tablaCurso){
128
129      /* Número de clases totales posibles (aproximación) */
130      const clases = 17;
131
132
133      let array = [{
134          label: "",
135          y: 0
136      }, {
137          label: "",
138          y: 0
139      }, {
140          label: "Clases totales",
141          y: clases
142      }
143      ];
144
145      await tablaCurso.then(data => {
146
147          /* Se construye el arreglo con dos objetos */
148
149          array[0].label = "Clases Realizadas"
150          array[0].y = parseInt(data[0].total_Horas)
151
152          array[1].label = "Clases no Realizadas"
153          array[1].y = clases - parseInt(data[0].total_Horas)
154
155      })
156
157      return array
158
159  }
160
```

img.5

```

162  async function getAprobados(tablaCalificaciones){
163
164      /* Se inicializa el arreglo de objetos */
165      let array = [
166          {
167              label: "Aprobados",
168              y: 0
169          },
170          {
171              label: "Desaprobados",
172              y: 0
173          },
174          {
175              label: "Sin nota",
176              y: 0
177          }
178      ]
179
180      await tablaCalificaciones.then(data => {
181          data.forEach(nota => {
182              /* NF: nota final, si esta es mayor a 11 esta aprobado */
183              if(nota.NF){
184                  if(nota.NF >= 11) array[0].y++
185                  else array[1].y++
186              }else{
187                  /* Sino no tiene nota final */
188                  array[2].y++;
189              }
190          })
191      })
192
193      return array
194  }

```

Img. 2

En esta parte se encuentra la función `getDatosCalificaciones` la cual tiene como parámetros la tabla de Clasificaciones y los Datos, en esta función vamos a obtener las calificaciones de los alumnos mediante su nombre y apellido . También está la función de las mejores notas y peores notas (`getLimitesCalificacioens`), en la cual están tres objetos la nota, mejor nota y peor nota

```

198  async function getDatosCalificaciones(tablaCalificaciones,tablaDatos){
199      let calificaciones = await tablaCalificaciones
200      let datos = await tablaDatos
201
202      calificaciones.forEach((alumno,index) => {
203          alumno["nombre"] = datos[index].nombre
204          alumno["apellido"] = datos[index].apellido
205
206      })
207
208      return calificaciones
209  }
210
211
212  /* Funcion para obtener mejor y peor nota */
213  async function getLimitesCalificaciones(tablaEstadistica){
214      let estadistica = await tablaEstadistica;
215
216      let array = [];
217      estadistica.forEach(nota => {
218          let obj = {}
219          obj["nota"] = nota.notas;
220          obj["mejorNota"] = [nota.mejorNota,nota.nomMejorNota];
221          obj["peorNota"] = [nota.peorNota,nota.nomPeorNota];
222
223          array.push(obj)
224      })
225
226      return array;
227  }

```

Img.6

e. Gráficos

```

1 function graficarBarras(data,div){
2     var chart = new CanvasJS.Chart(div, {
3         animationEnabled: true,
4         axisY: {
5             title: "Asistentes",
6             titleFontColor: "#4F81BC",
7             lineColor: "#4F81BC",
8             labelFontColor: "#4F81BC",
9             tickColor: "#4F81BC"
10        },
11        axisY2: {
12            title: "Faltos",
13            titleFontColor: "#C0504E",
14            lineColor: "#C0504E",
15            labelFontColor: "#C0504E",
16            tickColor: "#C0504E"
17        },
18        tooltip: {
19            shared: true
20        },
21        legend: {
22            cursor:"pointer",
23            itemclick: toggleDataSeries
24        },
25        data: [{
26            type: "column",
27            name: "Asistentes",
28            legendText: "Asistentes",
29            showInLegend: true,
30            dataPoints: data[0]
31        },
32        {
33            type: "column",
34            name: "Faltos",
35            legendText: "Faltos",
36            axisYType: "secondary",
37            showInLegend: true,
38            dataPoints: data[1]
39        }
40    ]});
41    chart.render();

```

```

83 graficarPie(totalAsistencia,"graficoTotalAsistencia")
84
85 /* Tabla y grafico de total de asistencia por clase */
86 const asistenciaPorClase = JSON.parse(document.getElementById("asistenciaPorClase").textContent),
87 tablaAsistenciaPorClase = document.getElementById("tablaAsistenciaPorClase").querySelector("tbody")
88
89
90 asistenciaPorClase.forEach(element => {
91   const tr = document.createElement("tr")
92   Object.values(element).forEach(attr => {
93     const td = document.createElement("td")
94     try{
95       attr = attr.replace(/_/g," ");
96     }catch(e){}
97     td.textContent = attr;
98     tr.appendChild(td)
99   })
100   tablaAsistenciaPorClase.appendChild(tr)
101 });
102
103
104 let arrayGlobal = [], presentes = [], faltos = []
105 asistenciaPorClase.forEach(element => {
106   let obj1 = {}, obj2 = {}
107   obj1.label = element.fecha.replace(/_/g," ");
108   obj1.y = element.presentes;
109
110   obj2.label = element.fecha.replace(/_/g," ");
111   obj2.y = element.faltos;
112
113   presentes.push(obj1)
114   faltos.push(obj2)
115 })
116
117 arrayGlobal.push(presentes)
118 arrayGlobal.push(faltos)
119
120 graficarBarras(arrayGlobal,"graficoAsistenciaPorClase");
121

```

f. Pdf

En esta parte se va a ver la sección de la creación del pdf

```

1 | document.getElementById('export').addEventListener('click', exportPDF);
2 |
3 | var specialElementHandlers = {
4 |   '.no-export': function(element, renderer) {
5 |     return true;
6 |   }
7 | };
8 |
9 | function exportPDF() {
10 |
11 |   var doc = new jsPDF('p', 'pt', 'a4');
12 |   var source = document.querySelectorAll(".content") //innerHTML;
13 |
14 |   var str = ''
15 |   source.forEach(el => str += el.innerHTML)
16 |
17 |
18 |   var margins = {
19 |     top: 10,
20 |     bottom: 10,
21 |     left: 10,
22 |     width: 595
23 |   };
24 |
25 |   doc.fromHTML(
26 |     str,
27 |     margins.left,
28 |     margins.top, {
29 |       'width': margins.width,
30 |       'elementHandlers': specialElementHandlers
31 |     },
32 |
33 |     function(dispose) {
34 |       doc.save('Test.pdf');
35 |     }, margins
36 |   );
37 |
38 | }

```

3.4 Calificaciones

En el archivo Calificaciones.php primeramente incluimos el archivo php que nos permitirá direccionarnos a las diferentes vistas funcionales disponibles según el tipo de sesión dado, llamado header.php, luego de lo cual tenemos dos accesos directos a través de botones con enlaces a vistas relacionadas como lo son la de ingresar notas y la generar nuevas notas de tipo subnota (dependientes de alguna nota continua). En el mismo archivo, podemos actualizar la vista a través enlazamiento con los datos enviados a ingresar notas a través del método POST, luego también tenemos en esta vista las funcionalidades para calcular las notas finales en un sistema de consulta a la base de datos, almacenamiento temporal de porcentajes de notas y retorno de valores en caso de que existan todas las notas necesarias para asignar una nota final. Estas consultas a la base de datos se usan también para mostrar todas las demás notas de los estudiantes del aula a través de arrays asociativos los cuales generan apuntadores a datos específicos para desplegar los datos de cada alumno en una respectiva columna asignada en pantalla según un ordenador de código u identificador que en este caso es la columna "cui".

```

1 <?php include "Includes/Header.php" ?>
2
3 <body>
4 <h2 style="text-align: center;">Estudiantes Registrados<br><br></h2>
5
6 <button onclick="location='Ingresar_Notas.php?clase=<?php echo $_GET['clase'] ?>'>Ingresar Notas</button>
7 <button onclick="location='Editar_Notas.php?clase=<?php echo $_GET['clase'] ?>'>Editar Notas</button>
8
9 <table id="tablaNotasEstudiantes" class="tabla">
10 <?php
11 $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
12 $BaseDatos->conectar();
13 $clase = $_GET["clase"];
14 $estudiantes = $BaseDatos->getEstudiantes($clase . "_datos");
15
16 echo "<thead>";
17 echo "<tr>";
18 echo "<th>CUI</th>";
19 echo "<th>Nombre</th>";
20 echo "<th>Apellidos</th>";
21
22 if(!is_null($estudiantes)) {
23     if ($_SERVER["REQUEST_METHOD"] == "POST") {
24         while ($row = mysqli_fetch_assoc($estudiantes)) {
25             $notas = $BaseDatos->getColumnasClases($clase . "_calificaciones");
26             if(!is_null($notas)) {
27                 while ($row_not = mysqli_fetch_assoc($notas)) {
28                     if ($row_not['column_name'] != 'cui' && $row_not['column_name'] != 'NF') {
29                         $campo = $row_not['column_name'];
30                         $cui = $row["cui"];
31                         if ($row_not['column_name'] == 'NC_1') {
32                             $camposxnot = $BaseDatos->getCamposNota($row_not['column_name'], $clase);
33                             if(!is_null($camposxnot)) {
34                                 $BaseDatos->insnota($clase, NULL, $campo, (int)$cui);
35                             } else {
36                                 $valor = $_POST[$row["cui"] . "_" . $row_not['column_name']];
37                                 if ($valor != NULL) {
38                                     $BaseDatos->insnota($clase, (float)$valor, $campo, (int)$cui);
39                                 } else {
40                                     $BaseDatos->insnota($clase, NULL, $campo, (int)$cui);
41                                 }
42                             }

```

```

39                                     $BaseDatos->insnota($clase, NULL, $campo, (int)$cui);
40                                 }
41                             }
42                         }
43                     }
44                 }
45             }
46         }
47     }
48 }
49 $estudiantes = $BaseDatos->getEstudiantes($clase . "_datos");
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }

```

3.5 Editar Notas

Una vez accesado este archivo, podemos ingresar valores y seleccionarlos con construcciones de html como lo son select e inputs, luego podemos realizar el envío de datos a través de AJAX usando el método POST dándose una validación de datos a través de un archivo javascript y enviando los datos para la modificación de tanto el informe como la base de datos para adecuarlos a la existencia de una nueva nota.

```
1 <?php include "Includes/Header.php" ?>
2
3 <body>
4 <h2 style="text-align: center;">Notas del Curso<br><br></h2>
5 <button onclick="location='Calificaciones.php?clase=<?php echo $_GET['clase'] ?>'>Volver</button>
6
7 <form>
8 <h2>Añadir nota</h2>
9
10 <div>
11 <label>Nombre de la nota</label>
12 <input id="nombrenewnota" type="text"/>
13 </div>
14 <div>
15 <label>Nota Superior</label>
16 <select name="notaSuperiornewnota" id="notaSuperiornewnota">
17 <option value="NC_1">NC_1</option>
18 <option value="NC_2">NC_2</option>
19 <option value="NC_3">NC_3</option>
20 </select>
21 </div>
22 <div>
23 <label>Porcentaje</label>
24 <select name="porcentajenewnota" id="porcentajenewnota">
25 <?php
26 for ($i = 1; $i <= 100; $i++) {
27     echo "<option value='$i'>$i</option>";
28 }
29 <?>
30 </select>
31 <button id="btnAgregar" type="button">Agregar</button>
32 </form>
33
34 <!-- <form method="post" action="Proyecto_Calificaciones.php?clase=<?php echo $_GET['clase'] ?>'> -->
35 <h2>Notas Existentes</h2>
36 <table id="tablaEstudiantes" class="tabla">
37 <thead>
38 <tr>
39 <th>Nota</th>
40 <th>Nota Superior</th>
41 <th>Porcentaje</th>
42 </tr>
43 </thead>
```

```
<h2>Notas Existentes</h2>
<table id="tablaEstudiantes" class="tabla">
<thead>
<tr>
<th>Nota</th>
<th>Nota Superior</th>
<th>Porcentaje</th>
</tr>
</thead>
<tbody id="contenido">
<?php
$BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
$BaseDatos->conectar();
$clase = $_GET["clase"];

$campnotas = $BaseDatos->getInfoNotas("notas","notaSuperior","porcentaje",$clase . "_informacion_y_estadistica");
if(!is_null($campnotas)) {
    while ($row = mysqli_fetch_assoc($campnotas)) {
        echo "<tr>";
        echo "<td>" . $row["notas"] . "</td>";
        echo "<td>" . $row["notaSuperior"] . "</td>";
        echo "<td>" . $row["porcentaje"] . "</td>";
        echo "</tr>";
    }
}
$BaseDatos->cerrar();
?>
</tbody>
</table>
<!-- <form type="submit" id="btnTomarAssist">Guardar</button>
-->
</form>
</body>
<script src="../../js/Proyecto_camb-notas.js"></script>
<!-- Archivo js necesario para mostrar el informe-->
<script src="../../js/getInforme.js"></script>
</html>
```


3.6 Ingresar Notas

Generamos un acceso a la vista de Calificaciones.php tanto así como una forma de enviar la nueva data que se genera según los inputs realizados en el archivo de tipo numéricos al mismo archivo mencionado usando el método POST. Especificando, en este archivo se generan apuntadores a espacios dentro de la base de datos donde se agregarán o cambiarán los valores indicados que por defecto y sin cambios son los de la misma base de datos. En caso de algunas notas, no podrán ser modificadas dado que si contienen notas derivadas, entonces su relleno depende de estas en medida según se especifica en los porcentajes asignados al generarlas.

```
1  <?php include "Includes/Header.php" ?>
2
3  <body>
4  <h2 style="text-align: center;">Estudiantes Registrados<br><br></h2>
5  <button onclick ="location='Calificaciones.php?clase=<?php echo $_GET['clase'] ?>'">Volver</button>
6  <form method="post" action="Proyecto_Calificaciones.php?clase=<?php echo $_GET['clase'] ?>">
7  <button type="submit" id="btnTomarAssist"/>Guardar</button>
8  <table id="tablaUsuarios" class="tabla">
9  <?php
10  $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
11  $BaseDatos->conectar();
12  $clase = $_GET['clase'];
13  $estudiantes = $BaseDatos->getEstudiantes($clase . "_datos");
14
15  echo "<thead>";
16  echo "<tr>";
17  echo "<th>CUI</th>";
18  echo "<th>Nombre</th>";
19  echo "<th>Apellidos</th>";
20
21  $notas = $BaseDatos->getColumnasClases($clase . "_calificaciones");
22  if(!is_null($notas)) {
23      while ($row = mysqli_fetch_assoc($notas)) {
24          if ($row['column_name'] != 'cui' && $row['column_name'] != 'NF') {
25              echo "<th> . $row['column_name'] . "</th>";
26          }
27      }
28  }
29
30  echo "</tr>";
31  echo "</thead>";
32  echo "<tbody>";
33
34  if(!is_null($estudiantes)) {
35      while ($row = mysqli_fetch_assoc($estudiantes)) {
36          echo "<tr>";
37          echo "<td class='CUI'>" . $row["cui"] . "</td>";
38          echo "<td class='nombre'>" . $row["nombre"] . "</td>";
39          echo "<td class='apellido'>" . $row["apellido"] . "</td>";
40
41          $notxest = $BaseDatos->getInfoEstudiantes($clase . "_calificaciones", $row["cui"]);
42          if(!is_null($notxest)) {
43              $row_nxe = mysqli_fetch_assoc($notxest);
44              $notas = $BaseDatos->getColumnasClases($clase . "_calificaciones");
45              while ($row_not = mysqli_fetch_assoc($notas)) {
46                  if ($row_not['column_name'] != 'cui' && $row_not['column_name'] != 'NF') {
47                      if ($row_not['column_name'] == 'NC_1') {
48                          $camposxnot = $BaseDatos->getCamposNota($row_not['column_name'], $clase);
```

```

34         if(!is_null($estudiantes)) {
35             while ($row = mysqli_fetch_assoc($estudiantes)) {
36                 echo "<tr>";
37                 echo "<td class='CUI'>" . $row["cui"] . "</td>";
38                 echo "<td class='nombre'>" . $row["nombre"] . "</td>";
39                 echo "<td class='apellido'>" . $row["apellido"] . "</td>";
40
41                 $notxest = $BaseDatos->getInfoEstudiantes($clase . "_calificaciones", $row["cui"]);
42                 if(!is_null($notxest)) {
43                     $row_nxe = mysqli_fetch_assoc($notxest);
44                     $notas = $BaseDatos->getColumnasClases($clase . "_calificaciones");
45                     while ($row_not = mysqli_fetch_assoc($notas)) {
46                         if ($row_not['column_name'] != 'cui' && $row_not['column_name'] != 'NF') {
47                             if ($row_not['column_name'] == 'NC_1') {
48                                 $camposxnot = $BaseDatos->getCamposNota($row_not['column_name'], $clase);
49                                 if(!is_null($camposxnot)) {
50                                     echo "<td>";
51                                     if ($row_nxe[$row_not['column_name']] != NULL) {
52                                         echo floatval($row_nxe[$row_not['column_name']]);
53                                     }
54                                     echo "</td>";
55                                 } else {
56                                     echo "<td><input name='". $row["cui"] . "_" . $row_not['column_name'] . "' value='". ""
57                                         type='text'></td>";
58                                 }
59                             } else if ($row_not['column_name'] == 'NC_2') {
60                                 $camposxnot = $BaseDatos->getCamposNota($row_not['column_name'], $clase);
61                                 if(!is_null($camposxnot)) {
62                                     echo "<td>";
63                                     if ($row_nxe[$row_not['column_name']] != NULL) {
64                                         echo floatval($row_nxe[$row_not['column_name']]);
65                                     }
66                                     echo "</td>";
67                                 } else {
68                                     echo "<td><input name='". $row["cui"] . "_" . $row_not['column_name'] . "' value='". ""
69                                         type='text'></td>";
70                                 }
71                             } else if ($row_not['column_name'] == 'NC_3') {
72                                 $camposxnot = $BaseDatos->getCamposNota($row_not['column_name'], $clase);
73                                 if(!is_null($camposxnot)) {
74                                     echo "<td>";
75                                     if ($row_nxe[$row_not['column_name']] != NULL) {
76                                         echo floatval($row_nxe[$row_not['column_name']]);
77                                     }
78                                     echo "</td>";
79                                 }
80                             }
81                         }
82                     }
83                 }
84             }
85         }
86     }
87 }

```

3.7 Portada

```

14 <html lang="es">
15 <head>
16 <meta charset="UTF-8"/>
17 <title>Clases</title>
18 <link href="https://fonts.googleapis.com/css?family=Poppins:600&display=swap" rel="stylesheet">
19 <link rel="stylesheet" type="text/css" href="css/portada_in.css">
20 <meta name="viewport" content="width=device-width, initial-scale=1">
21 </head>
22 <body>
23 <div class="container">
24 <div class="img">
25 
26 </div>
27 <div class="login-content">
28 <form action="Clase.php" method="get">
29 <h2 class="title">Bienvenido</h2>
30 <select name="clase" class="elegir">
31 <option value="Elegir clase">Elegir Clase</option>
32 </select>
33 </form>
34
35 $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
36 $BaseDatos->conectar();
37
38 $clases_profesor = $BaseDatos->getClases("semestre_actual");
39 $clases_codigo = $BaseDatos->getEncabezados("semestre_actual");
40
41 $i = 0;
42 if(!is_null($clases_codigo)){
43     while($row = mysqli_fetch_assoc($clases_codigo)){
44         $codigos[$i] = strtolower($row["Field"]);
45         $i = $i + 1;
46     }
47 }
48
49 $i = 1;
50 if(!is_null($clases_profesor)){
51     while ($row = mysqli_fetch_assoc($clases_profesor)) {
52         if($row["cui"] == $_SESSION["usuario"]){
53             foreach ($row as $key => $value) {
54                 if($value == 1){
55                     echo "<option value='". $codigos[$i] . "'>" . $BaseDatos->getNombreClase($codigos[$i]) . "</option>";
56                     $i = $i + 1;
57                 }
58             }
59         }
60     }
61 }

```

3.8 Clase

```

1  <?php include "Includes/Header.php" ?>
2
3  <body>
4
5      <h2 style="text-align: center;">Clase de
6      <?php
7          $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
8          $BaseDatos->conectar();
9
10         $clase = $BaseDatos->getNombreClase($_GET["clase"]);
11         echo "$clase";
12
13         $BaseDatos->cerrar();
14         ?><br><br>
15     </h2>
16
17     <button id="btn-informe" class="<?php $_GET['clase']?>">Informe</button>
18
19     <div class="capa"></div>
20 <!-- ----->
21
22     <table id="tablaUsuarios" class="tabla">
23         <h3 style="text-align: center;">Estudiantes Registrados</h3>
24         <thead>
25             <tr>
26                 <th>CUI</th>
27                 <th>Nombre</th>
28                 <th>Apellidos</th>
29             </tr>
30         </thead>
31         <tbody>
32             <?php
33                 /* Conexion con la base de datos */
34
35                 $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
36                 $BaseDatos->conectar();
37                 $codigo = $_GET["clase"];
38                 $estudiantes = $BaseDatos->getEstudiantes(strtolower($codigo));
39
40                 /* Se muestran los datos de los alumnos en una tabla */
41                 if(!is_null($estudiantes)) {
42                     while ($row = mysqli_fetch_assoc($estudiantes)) {
43                         echo "<tr>";
44                         echo "<td class='CUI'" . $row["cui"] . "</td>";
45                         echo "<td class='nombre'" . $row["nombre"] . "</td>";
46                         echo "<td class='apellido'" . $row["apellido"] . "</td>";
47                         echo "</tr>";
48                     }
49                 }
50             </?php>
51         </tbody>
52     </table>
53
54     </div>
55 </body>
56 </html>

```

4. ESTUDIANTE

5. LOGIN

5.1 Login.php (Portada)

```

1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF_8">
5      <title>Ingreso</title>
6      <link rel="stylesheet" href="../css/login.css">
7      <script type="text/javascript" src="login.js"></script>
8  </head>
9  <body>
10     <div class="container">
11         <div class="card"></div>
12         <div class="card">
13             <h1 class="title">Login</h1>
14             <form class="login-form">
15                 <div class="col_100">
16                     <input id="usuario" name="usuario" class="campos" type="text" placeholder="usuario"/>
17                 </div>
18                 <div class="col_100">
19                     <input id="clave" name="clave" class="campos" type="password" placeholder="clave"/>
20                 </div>
21                 <div class="col_100">
22                     <button id="btnAcceder" class="btn" type="button">ACCEDER</button>
23                 </div>
24             </form>
25             <p class="message_error" id="mensaje" name="mensaje"></p>
26         </div>
27     </div>
28
29 </body>
30 </html>

```

Img.1 Código Total login.php

En esta primera imagen 2 se visualiza la parte de la portada del Login, que tiene como título “Ingreso”, luego de esto se agrega dos enlaces para conectarlos con los estilos y con javascript (línea 6 y 7). El cuerpo de la página web está en una división llamada “container” (línea 10), este se subdivide en dos para dar el estilo de que parezca dos hojas. La segunda división “card” contiene otro título llamado Login (línea 13).

```

1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF_8">
5      <title>Ingreso</title>
6      <link rel="stylesheet" href="../css/login.css">
7      <script type="text/javascript" src="login.js"></script>
8  </head>
9  <body>
10     <div class="container">
11         <div class="card"></div>
12         <div class="card">
13             <h1 class="title">Login</h1>

```

Img.2

Además de esto, la división card se divide en tres partes más, Usuario, Clave, estas dos primeras son de elemento HTML <input> se usa para crear controles interactivos para formularios basados en la web con el fin de recibir datos del usuario. “Acceder” es un button con id “btnAcceder” que nos llevará dependiendo si es Profesor o alumno, esto se visualiza en la imagen 3.

```

4         <form class="login-form">
5             <div class="col_100">
6                 <input id="usuario" name="usuario" class="campos" type="text" placeholder="usuario"/>
7             </div>
8             <div class="col_100">
9                 <input id="clave" name="clave" class="campos" type="password" placeholder="clave"/>
10            </div>
11            <div class="col_100">
12                <button id="btnAcceder" class="btn" type="button">ACCEDER</button>
13            </div>
14        </form>

```

Img. 3

Y la parte final donde esta el párrafo Error, que nos mostrará si el usuario ha escrito mal su usuario y/o clave

```
23         </div>
24     </form>
25     <p class="message_error" id="mensaje" name="mensaje"></p>
26 </div>
27 </div>
28
29 </body>
30 </html>
```

Img. 4

5.2 login.js

```
const isEmpty = (str) => str.trim() === '';

function acceder() {
    var usua = document.getElementById('usuario');
    var clav = document.getElementById('clave');

    usuaVal = usua.value;
    clavVal = clav.value;

    if (isEmpty(usuaVal) || isEmpty(clavVal)) {
        alert('Todos los campos son obligatorios');
        return;
    }

    var contenido = document.getElementById("mensaje");

    if (window.XMLHttpRequest) {
        ajax = new XMLHttpRequest();
    } else {
        ajax = new ActiveXObject("Microsoft.XMLHTTP")
    }

    ajax.onreadystatechange = function() {
        if(ajax.readyState == 4 && ajax.status == 200) {
            if(ajax.responseText.trim() == "estudiante") {
                window.location.href = "../Estudiante/Portada.php";
            } else if(ajax.responseText.trim() == "docente") {
                ajax.open("POST", "../Profesor/Portada.php")
                window.location.href = "../Profesor/Portada.php";
            } else {
                contenido.innerHTML = ajax.responseText;
            }
        }
    }

    ajax.open("POST", "validar_login.php");
    ajax.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
    ajax.send("usuario=" + usuaVal + "&clave=" + clavVal);
};

function asignar() {
    btnAcceder = document.getElementById('btnAcceder');
    btnAcceder.addEventListener("click", acceder);
};

window.addEventListener("load", asignar);
```

img.1 Código Total login.js

En esta imagen se crearán dos funciones, acceder () y asignar().

La función asignar dentro tiene dos valores: usua (se obtiene el elemento mediante el ID “usuario”) y clave (se obtiene el elemento mediante el ID “clave”) como se ve en las líneas 4 y 5 . Se va a verificar que siempre se llenen esos campos, ecado de que no, aparecerá “Todos los campos son obligatorios” (10-13). Si vemos de la línea 23 hasta 34, ese es el proceso para saber donde vamos a enviar al

usuario, si es un estudiante la localización será “../Estudiante/Portada.php”, si es profesor será “../Profesor/Portada.php”. Todo esto se ve en la imagen 2.

```
3  function acceder() {
4      var usua = document.getElementById('usuario');
5      var clav = document.getElementById('clave');
6
7      usuaVal = usua.value;
8      clavVal = clav.value;
9
10     if (isEmpty(usuaVal) || isEmpty(clavVal)) {
11         alert('Todos los campos son obligatorios');
12         return;
13     }
14
15     var contenido = document.getElementById("mensaje");
16
17     if (window.XMLHttpRequest) {
18         ajax = new XMLHttpRequest();
19     } else {
20         ajax = new ActiveXObject("Microsoft.XMLHTTP")
21     }
22
23     ajax.onreadystatechange = function() {
24         if(ajax.readyState == 4 && ajax.status == 200) {
25             if(ajax.responseText.trim() == "estudiante") {
26                 window.location.href = "../Estudiante/Portada.php";
27             } else if(ajax.responseText.trim() == "docente") {
28                 ajax.open("POST", "../Profesor/Portada.php")
29                 window.location.href = "../Profesor/Portada.php";
30             } else {
31                 contenido.innerHTML = ajax.responseText;
32             }
33         }
34     }
35
36     ajax.open("POST", "validar_login.php");
37     ajax.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
38     ajax.send("usuario=" + usuaVal + "&clave=" + clavVal);
39 };
```

Img. 2

En la siguiente parte veremos la función de asignar, lo que hace es

```
41 function asignar() {
42     btnAcceder = document.getElementById('btnAcceder');
43     btnAcceder.addEventListener("click", acceder);
44 };
45
46 window.addEventListener("load",asignar);
47
```

Img. 3

5.3 validar_login.php

```

1  <?php
2  include("../Utils/base_datos.php");
3
4  session_start();
5
6  $BaseDatos = new base_datos("localhost", "root", "", "ti_ciencias_computacion");
7  $BaseDatos->conectar();
8
9  $usua = $_POST["usuario"];
10 $clav = $_POST["clave"];
11
12 $usuarios = $BaseDatos->getUsuarios();
13 $mensaje = "Mensaje que enviara";
14 if(!is_null($usuarios)) {
15     while ($row = mysqli_fetch_assoc($usuarios)) {
16         if ($row["usuario"] == $usua) {
17             if ($row["clave"] == md5($clav)) {
18                 $_SESSION["usuario"] = $usua;
19                 $BaseDatos->cerrar();
20                 $mensaje = $row["permisos"];
21                 echo $mensaje;
22                 return;
23             } else {
24                 $mensaje = "Clave Incorrecta";
25                 break;
26             }
27         } else {
28             $mensaje = "Usuario no encontrado";
29         }
30     }
31 } else {
32     $mensaje = "Usuario no encontrado";
33 }
34 echo $mensaje;
35 session_unset();
36 session_destroy();
37 $BaseDatos->cerrar();
38 ?>

```

6. ESTILOS

-Login

En esa parte se verán los estilos al cuerpo de la página, el fondo, la fuente y color de letra que tendrá "Roboto Draft", "Roboto". Se le pondrá estilo al mensaje de error, el cual las letras serán de color rojo. El estilo que tendrá el botón son:

- El color de letra será blanco
- Su posición es relativa, la diferencia con el margen de arriba es de
- El fondo de este botón será el color cuyo código es #A52019
- El radio de los bordes es de 25px, lo que hace es que se vean ovalados sus bordes.
- No tiene el borde (color).

```

1  * {
2      font-family: verdana, sans-serif;
3      text-align: center;
4      margin: 0px;
5      padding: 0px;
6  }
7
8  @charset "UTF-8";
9  body {
10     background: #e9e9e9;
11     color: #666666;
12     font-family: "RobotoDraft", "Roboto", sans-serif;
13     font-size: 14px;
14     -webkit-font-smoothing: antialiased;
15     -moz-osx-font-smoothing: grayscale;
16  }
17
18  .message_error {
19     color: red;
20  }
21
22  .btn {
23     position: relative;
24     display: block;
25     width: 300px;
26     left: 80px;
27     height: 50px;
28     border-radius: 25px;
29     outline: none;
30     border: none;
31     background-image: linear-gradient(to right, #A52019, #A52019, #A52019);
32     background-size: 200%;
33     font-size: 1.2rem;
34     color: #fff;
35     font-family: 'Poppins', sans-serif;
36     text-transform: uppercase;
37     margin: 1rem 0;
38     cursor: pointer;
39     transition: .5s;
40  }

```

Img. 1

En la siguiente parte está la clase “container” los estilos que entran son los siguientes:

- Márgenes de 100px
- Una diferencia entre de 100px con respecto a la altura de la pagina web

Container. car tendra


```

44 .container {
45     position: relative;
46     top: 100px;
47     max-width: 460px;
48     width: 100%;
49     margin: 0 auto 100px;
50 }
51 .container.active .card:first-child {
52     background: #f2f2f2;
53     margin: 0 15px;
54 }
55 .container.active .card:nth-child(2) {
56     background: #fafafa;
57     margin: 0 10px;
58 }
59 .container.active .card.alt {
60     top: 20px;
61     right: 0;
62     width: 100%;
63     min-width: 100%;
64     height: auto;
65     border-radius: 5px;
66     padding: 60px 0 40px;
67     overflow: hidden;
68 }
69 .container.active .card.alt .toggle {
70     position: absolute;
71     top: 40px;
72     right: -70px;
73     box-shadow: none;
74     transform: scale(10);
75     transition: transform 0.3s ease;
76 }
77 .container.active .card.alt .toggle:before {
78     content: "";
79 }
80 .container.active .card.alt .title,
81 .container.active .card.alt .input-container,
82 .container.active .card.alt .button-container {
83     left: 0;
84     opacity: 1;
85     visibility: visible;
86     transition: 0.3s ease;
87 }
88 .container.active .card.alt .title {
89     transition-delay: 0.3s;
90 }

```

La parte card tendrá los siguientes estilos:

- Una posición relativa.
- Radio del verode de 5px, forma ovalada.
- El fondo será blanco
- Car. container su contenido estará relativo con un espacio de 50px de la parte de abajo, unos márgenes de 0px 60px 0px 50px
- Car. input. container el contenido que tiene está en un posición relativa con un espacio de
- Car. title el color de la fuente de letra será del color #a52019, el espacio será de 32px

```

105 .card {
106   position: relative;
107   background: #ffffff;
108   border-radius: 5px;
109   padding: 60px 0 40px 0;
110   box-sizing: border-box;
111   box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
112   transition: 0.3s ease;
113 }
114 .card:first-child {
115   background: #fafafa;
116   height: 10px;
117   border-radius: 5px 5px 0 0;
118   margin: 0 10px;
119   padding: 0;
120 }
121 .card .title {
122   position: relative;
123   z-index: 1;
124   border-left: 5px solid #A52019;
125   margin: 0 0 35px;
126   padding: 10px 0 10px 50px;
127   color: #A52019;
128   font-size: 32px;
129   font-weight: 600;
130   text-transform: uppercase;
131 }
132 .card .input-container {
133   position: relative;
134   margin: 0 60px 50px;
135 }
136 .card .input-container input {
137   outline: none;
138   z-index: 1;
139   position: relative;
140   background: none;
141   width: 100%;
142   height: 60px;
143   border: 0;
144   color: #212121;
145   font-size: 24px;
146   font-weight: 400;
147 }

```

Al final la parte del Login se vera asi

