

# UNIVERSIDAD PUBLICA DE EL ALTO

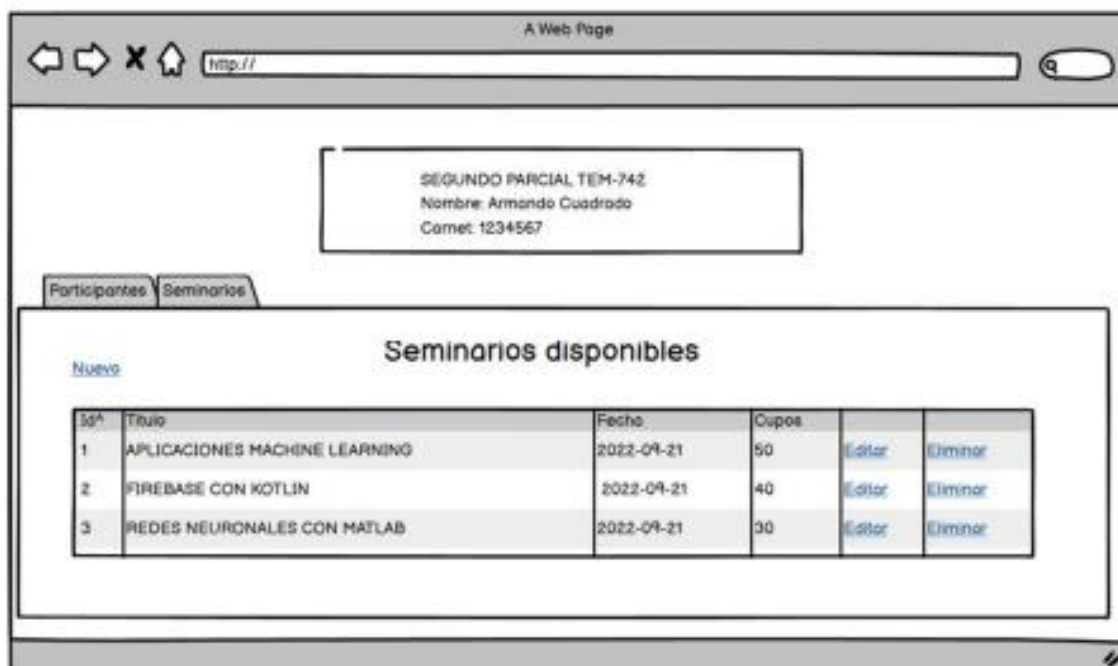
## INGENIERIA DE SISTEMAS



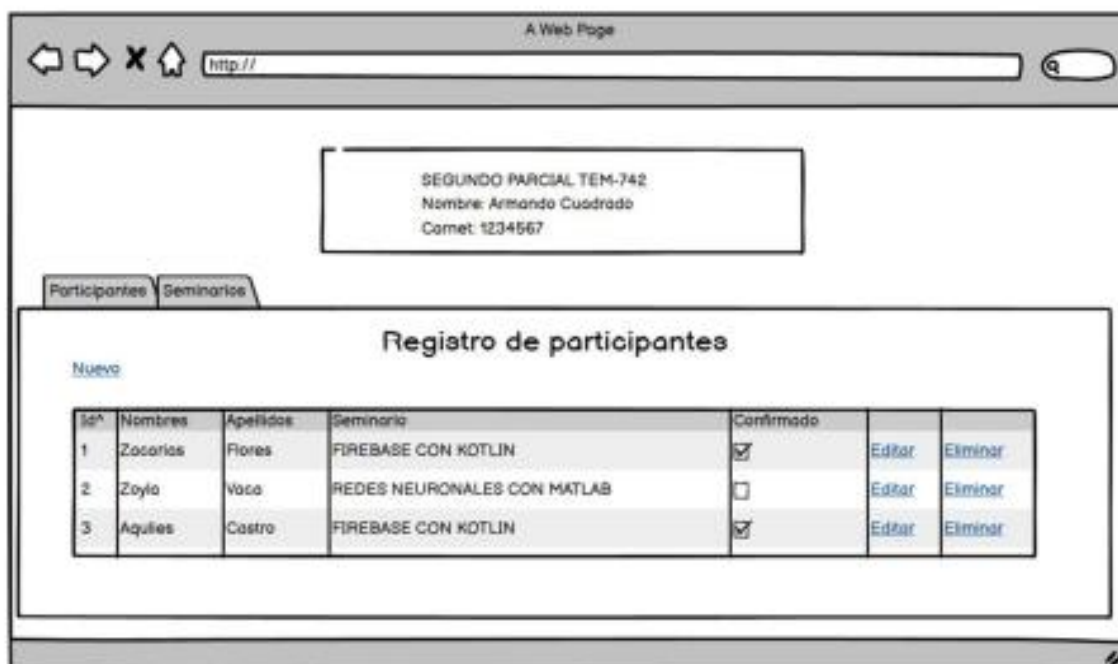
*Nombres: Cecilio Poma Muñoz*  
*Paralelo: 7C*  
*RU:200029033*  
*CI:7339585*  
*Celular:73018612*

*31 de Mayo de 2022*

1. Crear la base de datos **bd\_seminarios**
2. Ejecutar el script **bd\_seminarios.sql** (Si desea puede crear las tablas manualmente)
3. Crear el CRUD para la tabla **seminarios**



4. Crear el CRUD para la tabla **participantes**



Límite de entrega: 31 de mayo de 2022 Hrs: 22:30 en la plataforma virtual

Un archivo PDF con capturas de pantalla de la ejecución de la aplicación y el código fuente y 2) URL del proyecto en GitHub

```

1  CREATE DATABASE bd_seminarios CHARACTER SET 'utf8' COLLATE 'utf8_general_ci';
2
3  use bd_seminarios;
4
5  CREATE TABLE `seminarios` (
6      `id` int(11) NOT NULL AUTO_INCREMENT,
7      `titulo` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NULL DEFAULT NULL,
8      `fecha` varchar(10) CHARACTER SET utf8 COLLATE utf8_general_ci NULL DEFAULT NULL,
9      `cupo` int(11) NULL DEFAULT NULL,
10     PRIMARY KEY (`id`) USING BTREE
11 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT = DYNAMIC;
12
13
14 INSERT INTO `seminarios` VALUES (null, 'APLICACIONES MACHINE LEARNING', '2022-09-21', 50);
15 INSERT INTO `seminarios` VALUES (null, 'FIREBASE CON KOTLIN', '2022-09-21', 40);
16 INSERT INTO `seminarios` VALUES (null, 'REDES NEURONALES CON MATLAB', '2022-09-21', 30);
17
18
19 CREATE TABLE `participantes` (
20     `id` int(11) NOT NULL AUTO_INCREMENT,
21     `apellidos` varchar(50) CHARACTER SET utf8 COLLATE utf8_general_ci NULL DEFAULT NULL,
22     `nombres` varchar(60) CHARACTER SET utf8 COLLATE utf8_general_ci NULL DEFAULT NULL,
23     `id_seminario` int(11) NULL DEFAULT 1,
24     `confirmado` int(11) NULL DEFAULT 0,
25     PRIMARY KEY (`id`) USING BTREE,
26     INDEX `id_seminario`(`id_seminario`) USING BTREE,
27     CONSTRAINT `participantes_ibfk_1` FOREIGN KEY (`id_seminario`) REFERENCES
28     `seminarios` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
29 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci;
30
31
32 INSERT INTO `participantes` VALUES (null, 'Taxaca ', 'Cristian ', 1, 1);
33 INSERT INTO `participantes` VALUES (null, 'Calle', 'Daniel', 2, 0);
34 INSERT INTO `participantes` VALUES (null, 'Cutipa ', 'Richard ', 2, 1);
35 INSERT INTO `participantes` VALUES (null, 'Rocha ', 'Omar Bruno ', 3, 0);
36 INSERT INTO `participantes` VALUES (null, 'Silvestre ', 'Zenaida Kattia', 1, 1);
37 INSERT INTO `participantes` VALUES (null, 'Quispe', 'Wilmer', 2, 0);
38 INSERT INTO `participantes` VALUES (null, 'Nina', 'Gustavo', 3, 0);
39 INSERT INTO `participantes` VALUES (null, 'Perez', 'Juan', 3, 1);
40 INSERT INTO `participantes` VALUES (null, 'Aspi', 'Jhonny', 1, 1);
41 INSERT INTO `participantes` VALUES (null, 'Romero', 'Rita', 2, 1);
42
43 select * from participantes;
44
45

```

## ARCHIVOS JSP

### Jsp participantes-view

```
2  <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
3  <%@page contentType="text/html" pageEncoding="UTF-8"%>
4  <!doctype html>
5  <html lang="es">
6  <head>
7      <!-- Required meta tags -->
8      <meta charset="utf-8">
9      <meta name="viewport" content="width=device-width, initial-scale=1">
10     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
11         rel="stylesheet" >
12     <!-- Bootstrap CSS -->
13     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
14         rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBo
15         crossorigin="anonymous">
16
17     <title>Participantes</title>
18
19 </head>
20 <body class="main">
21
22     <div class="container">
23         <jsp:include page="META-INF/menu.jsp">
24             <jsp:param name="opcion" value="participantes"/>
25         </jsp:include>
26         <br>
27         <h1>Participantes</h1>
28         <br>
29         <a href="Controller_participantes?action=add" class="btn btn-primary btn-sm"><i class="fa
30         <br>
31         <br>
32         <table class="table table-info" border="1">
33
34             <tr>
35                 <th>Id</th>
36                 <th>Nombres</th>
37                 <th>Apellidos</th>
38                 <th>Seminarios</th>
39                 <th>Confirmado</th>
40                 <th>Editar</th>
41                 <th>Eliminar</th>
42             </tr>
```



```

43 <c:forEach var="conten" items="{cli}">
44 <tr>
45 <td>${conten.id}</td>
46 <td>${conten.nombres}</td>
47 <td>${conten.apellidos}</td>
48 <td>${conten.seminarios}</td>
49 <td>
50 <input class="chek" type="checkbox" name="#{conten.id}">
51 <c:if test="{conten.confirmado==1}" var="verificar" >
52 checked
53 </c:if>
54 >
55 </td>
56 <td><a href="Controller_participantes?action=edit&id=${conten.id}">
57 <i class="fa-solid fa-pencil"></i></a></td>
58 <td><a href="Controller_participantes?action=delete&id=${conten.id}">
59 onclick="return(confirm('Esta seguro de eliminar?'))" <i class="fa-re
60 </td>
61 </tr>
62 </c:forEach>
63 </table>
64 </div>
65
66 <!-- Java Scripts -->
67
68 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
69 integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
70 crossorigin="anonymous"></script>
71 <style>

```

```
72  .main{
73      background-image: url("img/informatica.jpg");
74      opacity: 10px;
75      background-position: center center;
76      background-size: cover;
77      height: auto;
78      left: 0;
79      min-height: 100%;
80      min-width: 100%;
81      margin: 0;
82      width: auto;
83      position: relative;
84  }
85
86  </style>
87  </body>
88  </html>
```

```

2  <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
3  <%@page contentType="text/html" pageEncoding="UTF-8"%>
4  <!doctype html>
5  <html lang="es">
6      <head>
7          <!-- Required meta tags -->
8          <meta charset="utf-8">
9          <meta name="viewport" content="width=device-width, initial-scale=1">
10         <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css"
11             rel="stylesheet" >
12         <!-- Bootstrap CSS -->
13         <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
14             rel="stylesheet"
15             integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
16             crossorigin="anonymous">
17
18         <title>registro </title>
19     </head>
20     <body class="main">
21         <div class="container">
22             <jsp:include page="META-INF/menu.jsp">
23                 <jsp:param name="opcion" value="participantes"/>
24             </jsp:include>
25             <br>
26             <h1>
27                 <c:if test="${requestScope.action=='nuevo'}" var="control" scope="request">
28                     Nuevo
29                 </c:if>
30                 <c:if test="${requestScope.action=='modificar'}" var="control" scope="request">
31                     Edicion de
32                 </c:if>
33                 Participantes
34             </h1>
35             <br>

```

```

36 <form action="Controller_participantes" method="post">
37   <input type="hidden" name="id" value="{par.id}">
38   <div class="mb-3">
39     <label for="" class="form-label">Nombres</label>
40     <input type="text" class="form-control" name="nombres"
41       value="{par.nombres}"placeholder="escriba su nombre" required>
42
43   </div>
44
45   <div class="mb-3">
46     <label for="" class="form-label">Apellidos</label>
47     <input type="text" class="form-control" name="apellidos"
48       value="{par.apellidos}" placeholder="escriba sus apellidos" required>
49   </div>
50
51   <div class="mb-3">
52     <label for="" class="form-label">Seminarios</label>
53     <select name="id_seminario" class="form-control" required >
54       <option value="">--Seleccione--</option>
55       <c:forEach var="valor" items="{list_seminario}">
56         <option value="{valor.id}"
57           <c:if test="{par.id_seminario==valor.id}" var="c" >
58             selected
59           </c:if>
60           >{valor.titulo}</option>
61       </c:forEach>
62     </select>
63   </div>
64
65   <div class="mb-3">
66     <label for="" class="form-label">Confirmar Participacion</label>
67
68     <input class="chek" type="checkbox" name="confirmado" value="1"
69       <c:if test="{par.confirmado=='1'}" var="verificar" >
70         checked

```



```

71         </c:if>
72     >
73
74     </div>
75
76     <button type="submit" class="btn btn-primary">Enviar</button>
77 </form>
78 <br>
79 <a href="Controller_participantes" class="btn btn-warning btn-sm">Volver</a>
80 <br>
81 <br>
82
83 </div>
84 <!-- Java Scripts -->
85
86 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
87     integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
88     crossorigin="anonymous"></script>
89 <style>
90     .main{
91         background-image: url("img/informatica.jpg");
92         opacity: 10px;
93         background-position: center center;
94         background-size: cover;
95         height: auto;
96         left: 0;
97         min-height: 100%;
98         min-width: 100%;
99         margin: 0;
100        width: auto;
101        position: relative;
102    }
103
104 </style>
105 </body>
106 </html>

```

```

1  <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
2
3  <%@page contentType="text/html" pageEncoding="UTF-8"%>
4  <!doctype html>
5  <html lang="es">
6      <head>
7          <!-- Required meta tags -->
8          <meta charset="utf-8">
9          <meta name="viewport" content="width=device-width, initial-scale=1">
10         <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css"
11             rel="stylesheet" >
12         <!-- Bootstrap CSS -->
13         <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
14             rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU
15             crossorigin="anonymous">
16
17         <title>Seminario</title>
18
19     </head>
20     <body class="main">
21
22         <div class="container">
23             <jsp:include page="META-INF/menu.jsp">
24                 <jsp:param name="opcion" value="seminarios"/>
25             </jsp:include>
26             <br>
27             <h1>Seminarios</h1>
28             <br>
29             <a href="Controller_seminarios?action=add" class="btn btn-primary btn-sm"><i class="f
30             <br>
31             <br>
32             <table class="table table-info" border="1">
33
34                 <tr>
35                     <th>Id</th>
36                     <th>Titulo</th>
37                     <th>Fecha</th>
38                     <th>Cupos</th>
39                     <th>Editar</th>
40                     <th>Eliminar</th>
41
42                 </tr>

```

```

43 <c:forEach var="conten" items="{list_seminario}">
44 <tr>
45 <td>{conten.id}</td>
46 <td>{conten.titulo}</td>
47 <td>{conten.fecha}</td>
48 <td>{conten.cupo}</td>
49 <td><a href="Controller_seminarios?action=edit&id={conten.id}"
50 ><i class="fa-solid fa-pencil"></i></a></td>
51 <td><a href="Controller_seminarios?action=delete&id={conten.id}"
52 onclick="return(confirm('Esta seguro de eliminar??'))" ><i class="fa-
53 </td>
54 </tr>
55 </c:forEach>
56 </table>
57 </div>
58
59 <!-- Java Scripts -->
60
61 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
62 integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
63 crossorigin="anonymous"></script>
64 <style>
65 .main{
66 background-image: url("img/informatica.jpg");
67 opacity: 10px;
68 background-position: center center;
69 background-size: cover;
70 height: auto;
71 left: 0;
72 min-height: 100%;
73 min-width: 100%;
74 margin: 0;
75 width: auto;
76 position: relative;
77 }
78
79 </style>
80 </body>
81 </html>

```

```

1  <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
2  <%@page contentType="text/html" pageEncoding="UTF-8"%>
3  <!doctype html>
4  <html lang="es">
5      <head>
6          <!-- Required meta tags -->
7          <meta charset="utf-8">
8          <meta name="viewport" content="width=device-width, initial-scale=1">
9          <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css"
10              rel="stylesheet" >
11          <!-- Bootstrap CSS -->
12          <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
13              rel="stylesheet"
14              integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
15              crossorigin="anonymous">
16
17          <title>registro </title>
18      </head>
19      <body class="main">
20          <div class="container">
21              <jsp:include page="META-INF/menu.jsp">
22                  <jsp:param name="opcion" value="seminarios"/>
23              </jsp:include>
24              <br>
25              <h1>
26                  <c:if test="${requestScope.action=='nuevo'}" var="control" scope="request">
27                      Nuevo
28                  </c:if>
29                  <c:if test="${requestScope.action=='modificar'}" var="control" scope="request">
30                      Edicion de
31                  </c:if>
32                      Seminario
33              </h1>
34              <br>
35              <form action="Controller_seminarios" method="post">
36                  <input type="hidden" name="id" value="${semi.id}">
37                  <div class="mb-3">
38                      <label for="" class="form-label">Titulo</label>
39                      <input type="text" class="form-control" name="titulo"
40                          value="${semi.titulo}"placeholder="Escriba su titulo del seminario" required>
41                  </div>
42                  <div class="mb-3">

```

```

43         <label for="" class="form-label">Fecha</label>
44         <input type="text" class="form-control" name="fecha"
45             value="{semi.fecha}" placeholder="Escriba fecha de registro" required>
46     </div>
47
48     <div class="mb-3">
49         <label for="" class="form-label">Cupos</label>
50         <input type="number" class="form-control" name="cupo"
51             value="{semi.cupo}" placeholder="escriba su N° de cupos" required >
52     </div>
53     <button type="submit" class="btn btn-primary">Enviar</button>
54 </form>
55 <br>
56 <a href="Controller_seminarios" class="btn btn-warning btn-sm">Volver</a>
57 <br>
58 <br>
59 </div>
60 <!-- Java Scripts -->
61 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
62     integrity="sha384-ka7Sk0Gln4gmtz2MlQnikTlwxGysOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
63     crossorigin="anonymous"></script>
64 <style>
65     .main{
66         background-image: url("img/informatica.jpg");
67         opacity: 10px;
68         background-position: center center;
69         background-size: cover;
70         height: auto;
71         left: 0;
72         min-height: 100%;
73         min-width: 100%;
74         margin: 0;
75         width: auto;
76         position: relative;
77     }
78 </style>
79 </body>
80 </html>

```



## Clases modelos - clase participantes

```
1  package com.codefix.jc.model;
2
3  public class Participantes {
4
5      private int id;
6      private String nombres;
7      private String apellidos;
8      private String seminarios;
9      private int confirmado;
10     private int id_semi;
11
12     public Participantes(int id, String nombres, String apellidos, String seminarios, int confirmado, int id_semi) {
13         this.id = id;
14         this.nombres = nombres;
15         this.apellidos = apellidos;
16         this.seminarios = seminarios;
17         this.confirmado = confirmado;
18         this.id_semi = id_seminario;
19     }
20
21     public Participantes() {
22         this.id = 0;
23         this.nombres = "";
24         this.apellidos = "";
25         this.seminarios = "";
26         this.confirmado = 0;
27         this.id_semi = 0;
28     }
29
30     public int getId() {
31         return id;
32     }
33
34     public void setId(int id) {
35         this.id = id;
36     }
37
38     public String getNombres() {
39         return nombres;
40     }
```

```
41
42 [-] public void setNombres(String nombres) {
43     |     this.nombres = nombres;
44     | }
45
46 [-] public String getApellidos() {
47     |     return apellidos;
48     | }
49
50 [-] public void setApellidos(String apellidos) {
51     |     this.apellidos = apellidos;
52     | }
53
54 [-] public String getSeminarios() {
55     |     return seminarios;
56     | }
57
58 [-] public void setSeminarios(String seminarios) {
59     |     this.seminarios = seminarios;
60     | }
61
62 [-] public int getConfirmado() {
63     |     return confirmado;
64     | }
65
66 [-] public void setConfirmado(int confirmado) {
67     |     this.confirmado = confirmado;
68     | }
69 [-] public int getId_seminario() {
70     |     return this.id_semi;
71     | }
72
73 [-] public void setId_seminario(int id_seminario) {
74     |     this.id_semi = id_seminario;
75     | }
76
77
78 }
79
```

```
1
2 package com.codefix.jc.model;
3
4 public class Seminarios {
5     private int id;
6     private String titulo;
7     private String fecha;
8     private int cupo;
9
10    public Seminarios(int id, String titulo, String fecha, int cupos) {
11        this.id = id;
12        this.titulo = titulo;
13        this.fecha = fecha;
14        this.cupo = cupos;
15    }
16
17    public Seminarios() {
18        this.id = 0;
19        this.titulo = "";
20        this.fecha = "";
21        this.cupo = 0;
22    }
23
24    public int getId() {
25        return id;
26    }
27
28    public void setId(int id) {
29        this.id = id;
30    }
31
32    public String getTitulo() {
33        return titulo;
34    }
35
```

```
36  public void setTitulo(String titulo) {
37      this.titulo = titulo;
38  }
39
40  public String getFecha() {
41      return fecha;
42  }
43
44  public void setFecha(String fecha) {
45      this.fecha = fecha;
46  }
47
48  public int getCupo() {
49      return cupo;
50  }
51
52  public void setCupo(int cupos) {
53      this.cupo = cupos;
54  }
55
56
57  }
58
```

```

1
2 package com.codefix.jc.controller;
3
4 import com.codefix.jc.dao.Participantes_dao;
5 import com.codefix.jc.dao.Participantes_dao_implements;
6 import com.codefix.jc.dao.Seminarios_dao;
7 import com.codefix.jc.dao.Seminarios_dao_implements;
8 import com.codefix.jc.model.Participantes;
9 import com.codefix.jc.model.Seminarios;
10 import java.io.IOException;
11 import java.io.PrintWriter;
12 import java.util.List;
13 import java.util.logging.Level;
14 import java.util.logging.Logger;
15 import javax.servlet.ServletException;
16 import javax.servlet.annotation.WebServlet;
17 import javax.servlet.http.HttpServlet;
18 import javax.servlet.http.HttpServletRequest;
19 import javax.servlet.http.HttpServletResponse;
20 import javax.swing.JOptionPane;
21
22
23 @WebServlet(name = "Controller_participantes", urlPatterns = {"/Controller_participantes"})
24 public class Controller_participantes extends HttpServlet {
25
26
27
28     @Override
29     protected void doGet(HttpServletRequest request, HttpServletResponse response)
30         throws ServletException, IOException {
31         try {
32             int id;
33             Participantes_dao dao = new Participantes_dao_implements();
34             Seminarios_dao dao_sem= new Seminarios_dao_implements();
35             Participantes c = new Participantes();
36
37             List<Seminarios> list=null;
38             String action = (request.getParameter("action") != null) ? request.getParameter("action") : null;
39             switch (action) {
40                 case "add":
41
42                 list = dao_sem.select();

```



```

43         request.setAttribute("action", "nuevo");
44         request.setAttribute("list_seminario", list);
45         request.setAttribute("par", c);
46         request.getRequestDispatcher("participantes-form.jsp").forward(request, response);
47         break;
48     case "edit":
49
50         list = dao_sem.select();
51         id=Integer.parseInt(request.getParameter("id"));
52         c=dao.getById(id);
53
54         request.setAttribute("action","modificar");
55         request.setAttribute("list_seminario", list);
56         request.setAttribute("par",c);
57         request.getRequestDispatcher("participantes-form.jsp").forward(request, response);
58         break;
59     case "delete":
60         id=Integer.parseInt(request.getParameter("id"));
61         dao.delete(id);
62         response.sendRedirect("Controller_participantes");
63         break;
64     case "view":
65         //obtener la lista de registros de clientes
66         List<Participantes> lista = dao.select();
67         request.setAttribute("cli", lista);
68         request.getRequestDispatcher("participantes-view.jsp").forward(request, response);
69         break;
70     }
71
72     } catch (Exception ex) {
73         JOptionPane.showMessageDialog(null, "Error: " + ex.getMessage());
74     }
75 }
76
77
78 @Override
79 protected void doPost(HttpServletRequest request, HttpServletResponse response)
80     throws ServletException, IOException {
81     int id =Integer.parseInt(request.getParameter("id"));
82     String nombres=request.getParameter("nombres");

```

```

83 String apellidos=request.getParameter("apellidos");
84 int id_seminario=Integer.parseInt(request.getParameter("id_seminario"));
85 String confirmado=request.getParameter("confirmado");
86 if(confirmado==null)
87 {
88     confirmado="0";
89 }
90 Participantes_dao dao=new Participantes_dao_implements();
91 Participantes cli= new Participantes ();
92 cli.setId(id);
93 cli.setNombres(nombres);
94 cli.setApellidos(apellidos);
95 cli.setId_seminario(id_seminario);
96 cli.setConfirmado(Integer.parseInt(confirmado));
97 if(id==0){
98     try {
99         dao.insert(cli);
100     } catch (Exception ex) {
101         Logger.getLogger(Controller_participantes.class.getName()).log(Level.SEVERE, null, ex);
102     }
103 }
104 else {
105
106     try {
107         //modificar registro
108         dao.update(cli);
109     } catch (Exception ex) {
110         JOptionPane.showMessageDialog(null,"ERROR :"+ex.getMessage());
111     }
112 }
113 }
114 response.sendRedirect("Controller_participantes");
115
116 }
117 }
118
119
120
121
122

```

```

1
2  package com.codefix.jc.controller;
3
4  import com.codefix.jc.dao.Participantes_dao;
5  import com.codefix.jc.dao.Participantes_dao_implements;
6  import com.codefix.jc.dao.Seminarios_dao;
7  import com.codefix.jc.dao.Seminarios_dao_implements;
8  import com.codefix.jc.model.Participantes;
9  import com.codefix.jc.model.Seminarios;
10 import java.io.IOException;
11
12 import java.util.List;
13 import java.util.logging.Level;
14 import java.util.logging.Logger;
15 import javax.servlet.ServletException;
16 import javax.servlet.annotation.WebServlet;
17 import javax.servlet.http.HttpServlet;
18 import javax.servlet.http.HttpServletRequest;
19 import javax.servlet.http.HttpServletResponse;
20 import javax.swing.JOptionPane;
21
22
23 @WebServlet(name = "Controller_seminarios", urlPatterns = {"/Controller_seminarios"})
24 public class Controller_seminarios extends HttpServlet {
25
26     @Override
27     protected void doGet(HttpServletRequest request, HttpServletResponse response)
28         throws ServletException, IOException {
29         try {
30             int id;
31             Seminarios_dao dao = new Seminarios_dao_implements();
32             Seminarios_dao dao_semi= new Seminarios_dao_implements();
33             Seminarios c = new Seminarios();
34
35
36             String action = (request.getParameter("action") != null) ? request.getParameter("action") : "view";
37             switch (action) {
38                 case "add":
39                     request.setAttribute("action", "nuevo");
40                     request.setAttribute("semi", c);
41                     request.getRequestDispatcher("seminarios-form.jsp").forward(request, response);
42                     break;
43                 case "update":
44                     request.setAttribute("action", "actualizar");
45                     request.setAttribute("semi", c);
46                     request.getRequestDispatcher("seminarios-form.jsp").forward(request, response);
47                     break;
48                 case "delete":
49                     request.setAttribute("action", "eliminar");
50                     request.setAttribute("semi", c);
51                     request.getRequestDispatcher("seminarios-form.jsp").forward(request, response);
52                     break;
53                 default:
54                     request.setAttribute("action", "view");
55                     request.getRequestDispatcher("seminarios-form.jsp").forward(request, response);
56                     break;
57             }
58         } catch (Exception e) {
59             e.printStackTrace();
60         }
61     }
62 }

```

```

43         case "edit":
44             id=Integer.parseInt(request.getParameter("id"));
45             c=dao.getById(id);
46             request.setAttribute("action","modificar");
47             request.setAttribute("semi",c);
48             request.getRequestDispatcher("seminarios-form.jsp").forward(request, response);
49             break;
50         case "delete":
51             id=Integer.parseInt(request.getParameter("id"));
52             dao.delete(id);
53             response.sendRedirect("Controller_seminarios");
54             break;
55         case "view":
56             //obtener la lista de registros de clientes
57             List<Seminarios> lista = dao.select();
58             request.setAttribute("list_seminario", lista);
59             request.getRequestDispatcher("seminarios-view.jsp").forward(request, response);
60             break;
61     }
62
63     } catch (Exception ex) {
64         JOptionPane.showMessageDialog(null, "Error: " + ex.getMessage());
65     }
66 }
67
68
69 @Override
70 protected void doPost(HttpServletRequest request, HttpServletResponse response)
71     throws ServletException, IOException {
72     int id =Integer.parseInt(request.getParameter("id"));
73     String titulo=request.getParameter("titulo");
74     String fecha=request.getParameter("fecha");
75     int cupo=Integer.parseInt(request.getParameter("cupo"));
76
77     Seminarios_dao dao=new Seminarios_dao_implements();
78     Seminarios cli= new Seminarios ();
79     cli.setId(id);
80     cli.setTitulo(titulo);
81     cli.setFecha(fecha);
82     cli.setCupo(cupo);

```

```

83
84     if(id==0){
85         try {
86             dao.insert(cli);
87         } catch (Exception ex) {
88             Logger.getLogger(Controller_participantes.class.getName()).log(Level.SEVERE, null, ex);
89         }
90     }
91     else {
92
93         try {
94             //modificar registro
95             dao.update(cli);
96         } catch (Exception ex) {
97             JOptionPane.showMessageDialog(null, "ERROR :"+ex.getMessage());
98
99         }
100     }
101     response.sendRedirect("Controller_seminarios");
102
103 }
104 }

```



## Interface DAO participantes

```
1
2 package com.codefix.jc.dao;
3 import com.codefix.jc.model.Participantes;
4
5 import java.util.List;
6
7
8 public interface Participantes_dao {
9     public void insert(Participantes participantes) throws Exception;
10    public void update(Participantes participantes) throws Exception;
11    public void delete(int id) throws Exception;
12    public Participantes getById(int id) throws Exception;
13    public List<Participantes> select() throws Exception;
14
15 }
```

## Clase DAO Participantes\_dao\_implements

```
1 package com.codefix.jc.dao;
2 import com.codefix.jc.util.Conection;
3 import com.codefix.jc.model.Participantes;
4 import java.sql.PreparedStatement;
5 import java.sql.ResultSet;
6 import java.util.ArrayList;
7 import java.util.List;
8 public class Participantes_dao_implements extends Conection implements Participantes_dao {
9     @Override
10    public void insert(Participantes participantes) throws Exception {
11        this.conectar();
12        String sql = "INSERT INTO participantes(nombres,apellidos,id_seminario,confirmado) VALUES(?,?,?,?)";
13        PreparedStatement ps = this.conn.prepareStatement(sql);
14        ps.setString(1, participantes.getNombres());
15        ps.setString(2, participantes.getApellidos());
16        ps.setInt(3, participantes.getId_seminario());
17        ps.setInt(4, participantes.getConfirmado());
18        ps.executeUpdate();
19        this.desconn();
20    }
21    @Override
22    public void update(Participantes participantes) throws Exception {
23        this.conectar();
24        String sql = "UPDATE participantes SET nombres=?,apellidos=?,id_seminario=?,confirmado=? WHERE id=?";
25
26        PreparedStatement ps = this.conn.prepareStatement(sql);
27        ps.setString(1, participantes.getNombres());
28        ps.setString(2, participantes.getApellidos());
29        ps.setInt(3, participantes.getId_seminario());
30        ps.setInt(4, participantes.getConfirmado());
31        ps.setInt(5, participantes.getId());
32        ps.executeUpdate();
33        this.desconn();
34    }
35    @Override
36    public void delete(int id) throws Exception {
37        this.conectar();
38        String sql = "DELETE FROM participantes WHERE id=" + id;
39        PreparedStatement ps = this.conn.prepareStatement(sql);
40        ps.executeUpdate();
41        this.desconn();
42    }
}
```

```

43
44     @Override
45     public Participantes getById(int id) throws Exception {
46         this.conectar();
47         Participantes cl = new Participantes();
48         String sql = "SELECT id,nombres,apellidos,id_seminario,confirmado FROM participantes WHERE id=" + id;
49         PreparedStatement ps = this.conn.prepareStatement(sql);
50         ResultSet rs = ps.executeQuery();
51         while (rs.next()) {
52             cl.setId(rs.getInt("id"));
53             cl.setNombres(rs.getString("nombres"));
54             cl.setApellidos(rs.getString("apellidos"));
55             cl.setId_seminario(rs.getInt("id_seminario"));
56             cl.setConfirmado(rs.getInt("confirmado"));
57         }
58         this.desconn();
59         return cl;
60     }
61
62     @Override
63     public List<Participantes> select() throws Exception {
64         List<Participantes> lista = new ArrayList<>();
65         this.conectar();
66         String sql = "SELECT P.id,P.nombres,P.apellidos,P.id_seminario,P.confirmado,S.titulo "
67             + "FROM participantes P INNER JOIN seminarios S ON (S.id=P.id_seminario)";
68         PreparedStatement ps = this.conn.prepareStatement(sql);
69         ResultSet rs = ps.executeQuery();
70         while (rs.next()) {
71             Participantes cl = new Participantes();
72             cl.setId(rs.getInt("id"));
73             cl.setNombres(rs.getString("nombres"));
74             cl.setApellidos(rs.getString("apellidos"));
75             cl.setId_seminario(rs.getInt("id_seminario"));
76             cl.setConfirmado(rs.getInt("confirmado"));
77             cl.setSeminarios(rs.getString("titulo"));
78             lista.add(cl);
79         }
80         this.desconn();
81         return lista;
82     }
83 }

```

## Interface DAO Seminarios\_dao

```
1 package com.codefix.jc.dao;
2
3 import com.codefix.jc.model.Seminarios;
4 import java.util.List;
5
6 public interface Seminarios_dao {
7
8     public void insert(Seminarios seminarios) throws Exception;
9
10    public void update(Seminarios seminarios) throws Exception;
11
12    public void delete(int id) throws Exception;
13
14    public Seminario getById(int id) throws Exception;
15
16    public List<Seminarios> select() throws Exception;
17 }
18
```

## Clase DAO Seminarios\_dao\_implements

```
1 package com.codefix.jc.dao;
2 import com.codefix.jc.model.Seminarios;
3 import com.codefix.jc.util.Conection;
4 import java.sql.PreparedStatement;
5 import java.sql.ResultSet;
6 import java.util.ArrayList;
7 import java.util.List;
8 public class Seminarios_dao_implements extends Conection implements Seminarios_dao{
9     @Override
10    public void insert(Seminarios seminarios) throws Exception {
11        this.conectar();
12        String sql = "INSERT INTO seminarios(titulo,fecha,cupo) VALUES(?,?,?)";
13        PreparedStatement ps = this.conn.prepareStatement(sql);
14        ps.setString(1, seminarios.getTitulo());
15        ps.setString(2, seminarios.getFecha());
16        ps.setInt(3, seminarios.getCupo());
17        ps.executeUpdate();
18        this.desconn();
19    }
20    @Override
21    public void update(Seminarios seminarios) throws Exception {
22        this.conectar();
23        String sql = "UPDATE seminarios SET titulo=?,fecha=?,cupo=? where id=?";
24        PreparedStatement ps = this.conn.prepareStatement(sql);
25
26        ps.setString(1, seminarios.getTitulo());
27        ps.setString(2, seminarios.getFecha());
28        ps.setInt(3, seminarios.getCupo());
29        ps.setInt(4, seminarios.getId());
30        ps.executeUpdate();
31        this.desconn();
32    }
33    @Override
34    public void delete(int id) throws Exception {
35        this.conectar();
36        String sql = "DELETE FROM seminarios WHERE id=" + id;
37        PreparedStatement ps = this.conn.prepareStatement(sql);
38        ps.executeUpdate();
39        this.desconn();
40    }
41    @Override
42    public Seminarios getById(int id) throws Exception {
```



```

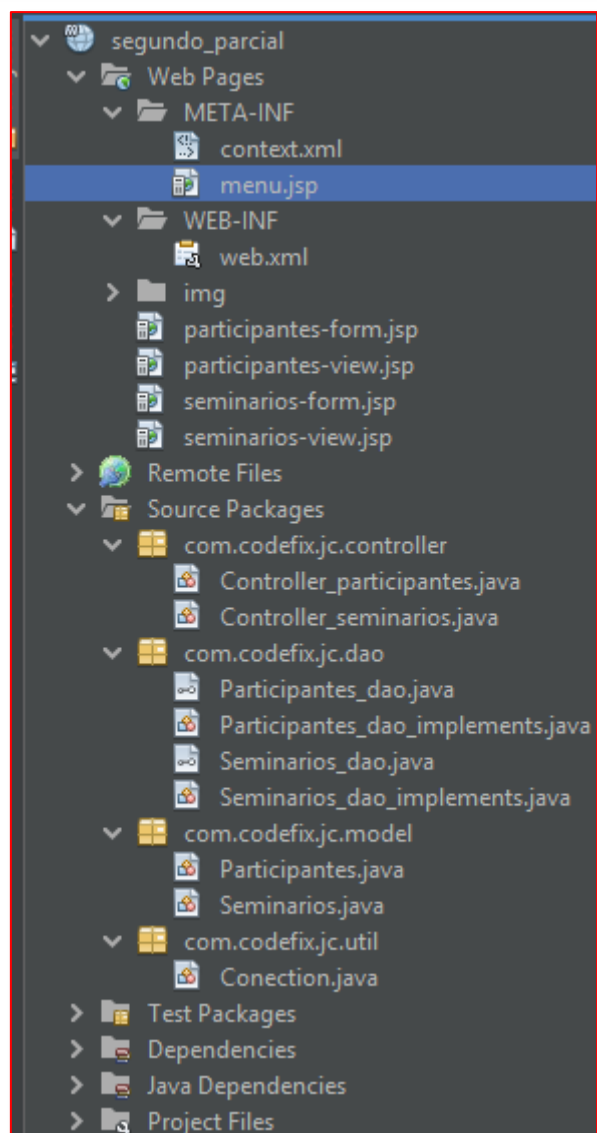
43     this.conectar();
44     Seminarios cl = new Seminarios();
45     String sql = "SELECT id,titulo,fecha,cupo FROM seminarios WHERE id=" + id;
46     PreparedStatement ps = this.conn.prepareStatement(sql);
47     ResultSet rs = ps.executeQuery();
48     while (rs.next()) {
49         cl.setId(rs.getInt("id"));
50         cl.setTitulo(rs.getString("titulo"));
51         cl.setFecha(rs.getString("fecha"));
52         cl.setCupo(rs.getInt("cupo"));
53     }
54     this.desconn();
55     return cl;
56 }
57
58 @Override
59 public List<Seminarios> select() throws Exception {
60     List<Seminarios> lista = new ArrayList<>();
61     this.conectar();
62     String sql = "SELECT id,titulo,fecha,cupo FROM seminarios";
63     PreparedStatement ps = this.conn.prepareStatement(sql);
64     ResultSet rs = ps.executeQuery();
65     while (rs.next()) {
66         Seminarios cl = new Seminarios();
67         cl.setId(rs.getInt("id"));
68         cl.setTitulo(rs.getString("titulo"));
69         cl.setFecha(rs.getString("fecha"));
70         cl.setCupo(rs.getInt("cupo"));
71         lista.add(cl);
72     }
73     this.desconn();
74     return lista;
75 }
76 }
77

```

```
1
2 package com.codefix.jc.util;
3
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6 import java.sql.SQLException;
7 import java.util.logging.Level;
8 import java.util.logging.Logger;
9 import javax.swing.JOptionPane;
10
11 public class Conexion {
12     static String driver="com.mysql.jdbc.Driver";
13     static String url="jdbc:mysql://localhost:3306/bd_seminarios";
14     static String usuario="root";
15     static String password="";
16     protected Connection conn=null;
17
18     public Conexion(){
19         try {
20             Class.forName(driver);
21             conn=DriverManager.getConnection(url,usuario,password);
22         } catch (Exception ex) {
23             JOptionPane.showMessageDialog(null,"Error del driver "+ex.getMessage());
24         }
25     }
26
27     public Connection conectar(){
28         return conn;
29     }
30
31     public void desconn()
32     {
33         try {
34             conn.close();
35         } catch (SQLException ex) {
36             Logger.getLogger(Conexion.class.getName()).log(Level.SEVERE, null, ex);
37         }
38     }
39 }
40
41 }
```

## Jsp menu cabecera

```
1 <%
2 //scriptles
3     String op = request.getParameter("opcion");
4 %>
5 <!-- -->
6 <ul class="nav nav-pills">
7 <li class="nav-item">
8     <a class="nav-link <%= (op.equals("participantes"))?"active":""%>"
9         aria-current="page" href="Controller_participantes">Participantes</a>
10 </li>
11 <li class="nav-item">
12     <a class="nav-link <%= (op.equals("seminarios"))?"active":""%>"
13         aria-current="page" href="Controller_seminarios">Seminarios</a>
14 </li>
15
16 <li class="nav-item">
17     <a class="nav-link disabled">Disabled</a>
18 </li>
19 </ul>
20
```



## Funcionamiento de la aplicación web

localhost:8080/segundo\_parcial/Controller\_seminarios

Aplicaciones YouTube virtual facebook documentos whatsapp conversor gitLab gitHub compilador speed

Participantes **Seminarios** Disabled

# Seminarios

+Nuevo

Id	Titulo	Fecha	Cupos	Editar	Eliminar
1	APLICACIONES MACHINE LEARNING	2022-09-21	50		
2	FIREBASE CON KOTLIN	2022-09-21	40		
3	REDES NEURONALES CON MATLAB	2022-09-21	30		
6	ARDUINO	30/05/2022	30		

localhost:8080/segundo\_parcial/

Aplicaciones YouTube virtual facebook documentos whatsapp conversor gitLab gitHub

**Participantes** Seminarios Disabled

# Participantes

+Nuevo

Id	Nombres	Apellidos	Seminarios	Confirmado	Editar	Eliminar
1	Cristian	Taxaca	APLICACIONES MACHINE LEARNING	<input type="checkbox"/>		
2	Daniel	Calle	FIREBASE CON KOTLIN	<input checked="" type="checkbox"/>		
4	Omar Bruno	Rocha	REDES NEURONALES CON MATLAB	<input type="checkbox"/>		
6	Wilmer	Quispe	FIREBASE CON KOTLIN	<input type="checkbox"/>		
7	Gustavo	Nina	REDES NEURONALES CON MATLAB	<input checked="" type="checkbox"/>		
8	Juan	Perez	REDES NEURONALES CON MATLAB	<input checked="" type="checkbox"/>		
9	Jhonny	Aspi	APLICACIONES MACHINE LEARNING	<input checked="" type="checkbox"/>		
10	Rita	Romero	FIREBASE CON KOTLIN	<input checked="" type="checkbox"/>		
11	jhonny	poma	FIREBASE CON KOTLIN	<input type="checkbox"/>		
12	jhonny yamil	poma	REDES NEURONALES CON MATLAB	<input checked="" type="checkbox"/>		
15	juana	poma	ARDUINO	<input checked="" type="checkbox"/>		

Nueva pestaña x registro x +

localhost:8080/segundo\_parcial/Controller\_participantes?action=add

Aplicaciones YouTube virtual facebook documentos whatsapp conversor gitLab gitHub

Participantes Seminarios Disabled

## Nuevo Participantes

Nombres

maria

Apellidos

poma

Seminarios

--Seleccione--

--Seleccione--

APLICACIONES MACHINE LEARNING

**FIREBASE CON KOTLIN**

REDES NEURONALES CON MATLAB

ARDUINO

Volver

localhost:8080/segundo\_parcial/Controller\_seminarios

Aplicaciones YouTube virtual facebook documentos whatsapp conversor gitLab gitHub

Participantes **Seminarios** Disabled

## Seminarios

+Nuevo

Id	Titulo	Fecha	Cupos	Editar	Eliminar
1	APLICACIONES MACHINE LEARNING	2022-09-21	50		
2	FIREBASE CON KOTLIN	2022-09-21	40		
3	REDES NEURONALES CON MATLAB	2022-09-21	30		
6	ARDUINO	30/05/2022	30		



localhost:8080/segundo\_parcial/Controller\_seminarios?action=add

Aplicaciones YouTube virtual facebook documentos whatsapp conversor gitLab gitHub

Participantes **Seminarios** Disabled

## Nuevo Seminario

Título

ROBOTICA

Fecha

31/05/2022

Cupos

15

Enviar

Volver

localhost:8080/segundo\_parcial/Controller\_seminarios











Aplicaciones YouTube virtual facebook documentos whatsapp conversor gitLab gitHub

Participantes **Seminarios** Disabled

localhost:8080 dice  
Esta seguro de eliminar??

Aceptar Cancelar

+ Nuevo

Id	Título	Fecha	Cupos	Editar	Eliminar
1	APLICACIONES MACHINE LEARNING	2022-09-21	50		
2	FIREBASE CON KOTLIN	2022-09-21	40		
3	REDES NEURONALES CON MATLAB	2022-09-21	30		
6	ARDUINO	30/05/2022	30		
7	ROBOTICA	31/05/2022	15		



localhost:8080/segundo\_parcial/Controller\_participantes

Aplicaciones YouTube gitLab gitHub

Participantes

Esta seguro de eliminar??

Aceptar Cancelar

+Nuevo

Id	Nombres	Apellidos	Seminarios	Confirmado	Editar	Eliminar
1	Cristian	Taxaca	APLICACIONES MACHINE LEARNING	<input type="checkbox"/>		
2	Daniel	Calle	FIREBASE CON KOTLIN	<input checked="" type="checkbox"/>		
4	Omar Bruno	Rocha	REDES NEURONALES CON MATLAB	<input type="checkbox"/>		
6	Wilmer	Quispe	FIREBASE CON KOTLIN	<input type="checkbox"/>		
7	Gustavo	Nina	REDES NEURONALES CON MATLAB	<input checked="" type="checkbox"/>		
8	Juan	Perez	REDES NEURONALES CON MATLAB	<input checked="" type="checkbox"/>		
9	Jhonny	Aspi	APLICACIONES MACHINE LEARNING	<input checked="" type="checkbox"/>		
10	Rita	Romero	FIREBASE CON KOTLIN	<input checked="" type="checkbox"/>		
11	jhonny	poma	FIREBASE CON KOTLIN	<input type="checkbox"/>		
12	jhonny yamil	poma	REDES NEURONALES CON MATLAB	<input checked="" type="checkbox"/>		
15	juana	poma	ARDUINO	<input checked="" type="checkbox"/>		
16	maria	poma	FIREBASE CON KOTLIN	<input checked="" type="checkbox"/>		