

[<<CLICK PARA VER INDICE>>](#)

CASOS DE PRUEBA ASOCIADAS AL USUARIO VISITANTE

F01 - Ingreso a la página

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F01	Ingreso a página	Cargar página	-	Se visualiza página principal	Se visualiza página principal	Pasa

F02 – Crear un usuario <<[CLICK PARA VER INDICE](#)>>

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F02 - Caso1	Ingresar a interfaz “crear usuario”	1- Click en “Crear usuario docente”	-	Visualizar interfaz para crear usuario docente	Deberá ver los campos: Nombre, usuario, contraseña y boton ingresar	Pasa
F02 - Caso2	Ingresar a interfaz “crear usuario”	1- Click en “Crear usuario alumno”	-	Visualizar interfaz para crear usuario alumno	Deberá ver los campos: Nombre, usuario, contraseña, desplegable de docentes y boton ingresar	Pasa
F02 - Caso3	Ingresar a interfaz “crear usuario”	1- Click en “Crear usuario alumno” / “Crear usuario docente” 2 - Ingresa nombre Usuario inexistente 3- Contraseña válida	(ejemplo docente) Nombre: Luis Usuario: lu60 Contraseña: Frd60 (ejemplo alumno) Nombre: Pablo Usuario: pab32 Contraseña: tR32i Docente: jos45	Visualizar interfaz que corresponda según tipo de usuario	Visualizar botones que correspondan según tipo de usuario	Pasa
F02 - Caso4	Ingresar a interfaz “crear usuario”	1- Click en “Crear usuario alumno” / “Crear usuario docente” 2 - Ingresa datos en los campos. Alguno de los campos vacío	(ejemplo docente) Nombre: Luis Usuario: Contraseña: Frd60 (ejemplo alumno) Nombre: Pablo Usuario: pab312 Contraseña: Docente: jos45	Mensaje de error No permite continuar	Se vacían los campos	Falla
F02 - Caso5	Ingresar a interfaz “crear usuario”	1- Click en “Crear usuario alumno” / “Crear usuario docente” 2 - Ingresa nombre Usuario inexistente 3- Contraseña sin número o mayúscula	(ejemplo docente) Nombre: Luis Usuario: lu88 Contraseña: frd60 (ejemplo alumno) Nombre: Pablo Usuario: pab312 Contraseña: pabJ Docente: jos45	Mensaje de error No permite continuar	Se vacían los campos	Falla

F03 – Iniciar sesión <<[CLICK PARA VER INDICE](#)>>

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F03 - Caso1	Iniciar sesión	1- Pongo usuario 2- Pongo contraseña 3 - Ingreso	Usuario: jos45 Contraseña: Asd4	Visualizar interfaz de usuario docente	Deberá ver los botones correspondientes a su tipo de usuario	Pasa
F03 - Caso2	Iniciar sesión	1- Completo un solo campo 2 - Ingreso	Usuario: jos45 Contraseña: / Usuario: Contraseña:Asd4	Mensaje de error no permite continuar	Se vacía los campos	Falla
F03 - Caso3	Iniciar sesión	1 - Campos vacíos	-	Mensaje de error no permite continuar	Se vacía los campos	Falla

CASOS DE PRUEBA ASOCIADAS AL USUARIO DOCENTE

F04 – Plantear ejercicio [<<CLICK PARA VER INDICE>>](#)

Precondición : Click en botón "Plantear ejercicio"

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F04 - Caso1	Plantear ejercicio	1- Ingreso título 2- Subo imagen 3 - Ingreso descripción con más de 20 y menos de 200 caracteres	-	Mensaje de creación exitosa	Se crea un nuevo ejercicio. Se vacían los campos	Pasa
F04 - Caso2	Plantear ejercicio	1- Dejo campos vacíos 2 - Ingreso	-	Mensaje de error no permite continuar	Se vacía los campos	Falla
F04 - Caso3	Plantear ejercicio	1- Ingreso título 2- Subo imagen 3 - Ingreso descripción con menos de 20 y/o más de 200 caracteres	-	Mensaje de error no permite continuar	Se vacía los campos	Falla

F05 – Asignar nivel <<[CLICK PARA VER INDICE](#)>>

Precondición : Docente logueado

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F05 - Caso1	Asignar nivel	1 - Seleccionar un alumno en nivel "inicial" o "intermedio" 2 - Presionar "ascender"	-	Se actualizará la tabla mostrando el alumno con su nivel actualizado	El alumno ahora sólo podrá ver los ejercicios que correspondan a su nuevo nivel	Pasa
F05 - Caso2	Asignar nivel	1 - Seleccionar un alumno en nivel "avanzado" 2 - Presionar "ascender"	-	No sucederá nada	La tabla permanecerá sin alterar	Falla

F06 – Generar devoluciones <<[CLICK PARA VER INDICE](#)>>

Precondición : Docente logueado como: Usuario (jos45) - Contraseña (Asd4)

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F06 - Caso1	Generar devoluciones	1 - Seleccionar el ejercicio "E mayor(alumno1)" 2 - No ingreso datos en el campo de devolución	-	Mensaje de error no permite continuar	Sigue visible el ejercicio a calificar	Falla
F06 - Caso2	Generar devoluciones (ejercicio ya calificado)	1 - Seleccionar el ejercicio "E menor(alumno1)"	-	Mensaje de error no permite continuar	Seguirá visible el combo box para una nueva selección	Falla
F06 - Caso3	Generar devoluciones	1 - Seleccionar el ejercicio "E mayor(alumno1)" 2 - Redactar una devolución	Texto aleatorio	Se oculta la entrega calificada. Combo box visible para una nueva selección	Se genera la devolución. El alumno ya podrá verla al loguearse y consultar sus devoluciones	Pasa

F07 – Información estadística <<[CLICK PARA VER INDICE](#)>>

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F07 - Caso1	Estadísticas	-	-	Visualizar: total de ejercicios realizados para ese docente, combo box de alumnos para ver info de entregas y tabla de alumnos que realizaron más ejercicios	Datos correspondientes al alumno seleccionado. La interfaz será igualmente visible aunque no hayan datos	Pasa
F07 - Caso2	Estadísticas	1 - Seleccionar uno de los alumnos	-	Datos correspondientes al alumno seleccionado. La interfaz será igualmente visible aunque no hayan datos	Datos correspondientes al alumno seleccionado. Combo box de alumnos para ver info de entregas y tabla de alumnos que realizaron más ejercicios	Pasa

CASOS DE PRUEBA ASOCIADAS AL USUARIO ALUMNO

F08 – Buscar ejercicios [<<CLICK PARA VER INDICE>>](#)

Precondición : Alumno logueado como: Usuario (alumno29) - Contraseña (Asd4)

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F08 - Caso1	Buscar ejercicios (Coincidencia por título)	1 - Ingresar "old" 2 - Buscar	Texto tipeado: "old" (minúscula)	Visualizar el ejercicio titulado como: "Old McDonald"	Se lista el ejercicio encontrado	Pasa
F08 - Caso2	Buscar ejercicios (Coincidencia por descripción)	1 - Ingresar "realizar" 2 - Buscar	Texto tipeado: "realizar" (minúscula)	Visualizar los ejercicios titulados como: "E major - pre avanzado" y "Combinado en Do"	Se listan los ejercicios encontrados	Pasa
F08 - Caso3	Buscar ejercicios (Sin coincidencias)	1 - Ingresar "hola" 2 - Buscar	Texto tipeado: "hola"	Muestra un mensaje de error: "No hay resultados que coincidan con su búsqueda"	Muestra un mensaje de error y vacía el campo de búsqueda	Falla
F08 - Caso4	Buscar ejercicios (Campo vacío)	1 - Buscar	-	Muestra un mensaje de error: "Error, el campo no puede estar vacío"	Muestra un mensaje de error: "Error, el campo no puede estar vacío"	Falla

F09 – Entregar ejercicios <<[CLICK PARA VER INDICE](#)>>

Precondición : Alumno logueado como: Usuario (alumno29) - Contraseña (Asd4)

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F09 - Caso1	Entregar ejercicios	1 - Seleccionar "Old McDonald" 2 - Presionar el boton Seleccionar 3 - Adjuntar un archivo de sonido 4- Entregar	-	Se oculta el ejercicio. Continúa visible el combo box	Se genera una entrega	Pasa
F09 - Caso2	Entregar ejercicios	1 - Seleccionar "Combinado en Do" 2 - Presionar el botón Seleccionar	-	Muestra un mensaje de error: "Este ejercicio ya fue enviado para calificar"	No se genera entrega Continúa visible el combo box	Falla
F09 - Caso3	Entregar ejercicios	1 - Seleccionar "La marcha de los santos" 2 - Presionar el boton Seleccionar 3 - No adjuntar un archivo 4- Entregar	-	Muestra un mensaje de error: "Error, se debe adjuntar un archivo para la entrega"	No se genera entrega Continúa visible el combo box	Falla

F10 –Ver ejercicios resueltos <<[CLICK PARA VER INDICE](#)>>

Precondición : Alumno logueado como: Usuario (alumno1) - Contraseña (Asd4)

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F10 - Caso1	Ver ejercicios resueltos	1 - Seleccionar “Ver ejercicios resueltos”	-	Se listan los ejercicios resueltos por el alumno. Nombre, estado y fallo	Se listan los ejercicios resueltos por el alumno. Nombre, estado y fallo	Pasa

F11 – Información estadística <<[CLICK PARA VER INDICE](#)>>

Precondición : Alumno logueado como: Usuario (alumno2) - Contraseña (Asd4)

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F11 - Caso1	Información estadística	1 - Seleccionar "Información estadística"	-	Mostrar: Ha resuelto el 40% de las propuestas para su nivel. 0 de las entregas han sido calificadas 2 de las entregas están pendientes de calificación	Mostrar: Ha resuelto el 40% de las propuestas para su nivel. 0 de las entregas han sido calificadas 2 de las entregas están pendientes de calificación	Pasa

F12 – Cerrar sesión <<[CLICK PARA VER INDICE](#)>>

ID	Escenario del test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado: (F = falla, P = pasa)
F12 - Caso1	Cerrar sesión	1 - Seleccionar “Cerrar sesión” del menú auxiliar	-	Visualizar página principal	Se vacía el usuario logueado	Pasa

INDICE

[F01 – Ingreso a la página – Usuario/s: Visitante – Docente - Alumno](#)

[F02 – Crear un usuario – Usuario/s: Visitante](#)

[F03 – Iniciar sesión – Usuario/s: Visitante, Docente, Alumno](#)

[F04 – Plantear ejercicio – Usuario/s: Docente](#)

[F05 – Asignar nivel – Usuario/s: Docente](#)

[F06 – Generar devoluciones – Usuario/s: Docente](#)

[F07 – Información estadística – Usuario/s: Docente](#)

[F08 – Buscar ejercicios – Usuario/s: Alumno](#)

[F09 – Entregar ejercicios – Usuario/s: Alumno](#)

[F10 – Ver ejercicios resueltos – Usuario/s: Alumno](#)

[F11 – Información estadística – Usuario/s: Alumno](#)

[F12 – Cerrar sesión – Usuario/s: Docente, Alumno](#)

-----CODIGO COMPLETO-----

---HTML---

```
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="estilos.css">
<meta name="author" content="Maximiliano Gambetta">
<title>Document</title>
</head>
<body>
<div id="contenedor">
<div id="cabezal" class="clearfix">

<ul id="navPrincipal">
<!-- botones de funcionalidad para usuario tipo docente-->
<li id="btnCrearDocente">
<a>Crear usuario docente</a>
</li>
<li id="btnCrearAlumno">
<a>Crear usuario alumno</a>
</li>

<li id="btnPlantearEjercicio">
<a>Plantear ejercicio</a>
</li>

<li id="btnAsignarNivel">
```

```
<a>Asignar nivel</a>
</li>
```

```
<li id="btnDevoluciones">
  <a>Devoluciones</a>
</li>
```

```
<li id="btnInformacionEstadistica">
  <a>Información estadística</a>
</li>
```

```
<!-- botones de funcionalidad para usuario tipo alumno-->
```

```
<li id="btnEjerciciosPendientes">
  <a>Buscar ejercicios</a>
</li>
```

```
<li id="btnRealizarEntrega">
  <a>Entregar ejercicio</a>
</li>
```

```
<li id="btnEjerciciosResueltos">
  <a>Ejercicios resueltos</a>
</li>
```

```
<li id="btnInfoEstadistica">
  <a>Información estadística</a>
</li>
```

```
</ul>
```

```
<div id="aux">
```

```
<a id="btnIniciarSesion">Iniciar sesión</a>
<a id="btnCerrarSesion">Cerrar sesión</a>
```

```
</div><!-- FIN AUX -->
</div><!-- FIN CABEZAL -->
<div id="contenido">
```

```
<!-- div crear usuario -->
<div id="divCrearUsuario">
  <form id="formRegistrar" style="border-style: inset;">
    <label for="txtNombre">Nombre</label>
    <input type="text" id="txtNombre"> <br>

    <label for="txtUsuario">Usuario</label>
    <input type="text" id="txtUsuario"> <br>

    <label for="txtPass">Contraseña</label>
    <input type="text" id="txtPass"> <br>

    <p id="tituloDocente">Seleccione el docente</p>
    <select id="slcprofe">

    </select> <br><br>

    <input type="button" value="Registrar" id="btnRegistrar">

  </form>
</div>
```

```
<!-- div iniciar sesion -->
<div id="divIniciarSesion">
  <form id="formLogin" style="border-style: inset; text-align: right;">

```



```
<label for="txtuser">Usuario</label>
<input type="text" id="txtuser"> <br>
```

```
<label for="txtcontra">Contraseña</label>
<input type="text" id="txtcontra"> <br>
<input type="button" value="Iniciar sesión" id="btnLogueo">
</form>
</div>
```

```
<!--
DIVS DE USUARIO TIPO DOCENTE
-->
```

```
<!-- div de asignacion de niveles-->
<div id="divAsignarNivel">
  <form style="border-style: inset;">
    <label for="slcAscenderAlumno" style="font-size: 12px;">Seleccionar <br> alumno</label>
    <select id="slcAscenderAlumno">
    </select>
```

```
<input type="button" value="Ascender" id="btnAscensoAlumno">
</form>
```

```
<form style="border-style: inset;">
<table border="1" style="width: 100%;">
  <tr>
    <th>
      Nombre
    </th>
    <th>
```

Usuario

</th>

<th>

Nivel

</th>

</tr>

<tbody id="tablaNiveles">

</tbody>

</table>

</form>

</div>

<!-- div para planteo de ejercicios -->

<div id="divGenerarEjercicio">

<form style="border-style: inset;">

<h3>Ejercicio</h3>

<label for="txtTituloEjercicio">Título</label>

<input type="text" id="txtTituloEjercicio">

</form>

<form style="border-style: inset;">

<label for="foto">Cargar imagen</label>

<input type="file" name="foto" id="foto">

</form>

<form style="border-style: inset;">

<label for="txtDescripcion">Descripción:</label>

<textarea id="txtDescripcion" style="width: 100%;"></textarea>

</form>

<form style="border-style: inset;">

```
<label for="slcNivelDeEjercicio">Dificultad</label>
<select id="slcNivelDeEjercicio">
  <option value="Inicial">Inicial</option>
  <option value="Intermedio">Intermedio</option>
  <option value="Avanzado">Avanzado</option>
</select>
</form> <br>
```

```
<input type="button" value="Crear" id="btnCrearEjercicio"><br>
</form>
</div>
```

```
<!-- div para calificar ejercicios -->
<div id="divDevoluciones">
  <form style="border-style: inset;">
    <label for="slcDevoluciones" style="font-size: 12px;">Seleccionar <br> entrega</label>
    <select id="slcDevoluciones">
      <option value="Incompleto">Incompleto</option>
      <option value="Completado">Completado</option>
    </select>
```

```
<input type="button" value="Seleccionar" id="btnSeleccionarEntrega">
  </form>
  <div id="formularioDeCalificacion">
    <p id="pEntrega"></p>
    <form style="border-style: inset;">
      <textarea id="txtDevolucion"></textarea><br>
      <input type="button" value="Enviar" id="btnEnviarDevolucion" >
    </form>
  </div>
</div>
```

```
<!-- div informacion estadística docente -->
```

```
<div id="divEstadisticasDeDocente">
```

```
<p id="pTotalEntregas"></p>
```

```
<form style="border-style: inset;">
```

```
<label for="slcAlumnosDeEseProfe" style="font-size: 12px;">Seleccione a uno <br> de sus alumnos</label>
```

```
<select id="slcAlumnosDeEseProfe">
```

```
</select>
```

```
<input type="button" value="Seleccionar" id="btnDocEstadistica">
```

```
<p id="pCantidadEjerciciosPorNivel"></p>
```

```
<p id="pCantidadEjerciciosResueltos"></p>
```

```
</form>
```

```
<form style="border-style: inset;">
```

```
<h3>Alumnos que han resuelto más ejercicios</h3>
```

```
<table border="1" style="width: 100%;">
```

```
<tr>
```

```
<th>
```

```
Nombre
```

```
</th>
```

```
<th>
```

```
Usuario
```

```
</th>
```

```
<th>
```

```
Total
```

```
</th>
```

```
</tr>
```

```
<tbody id="alumnoQueResolvioMas">
```

```

    </tbody>
</table>
</form>

```

```
</div>
```

```
<!--
DIVS DE USUARIO TIPO ALUMNO
-->
```

```
<!-- div de busqueda de ejercicios -->
<div id="divBuscarEjercicio">
    <form style="border-style: inset;">
        <label for="txtBuscarEjercicio">Buscar</label>
        <input type="text" id="txtBuscarEjercicio">
        <input type="button" value="Buscar" id="btnBusquedaDeEjercicio">
    </form>
    <table border="1" style="width: 100%;">
        <tr>
            <th width="30%">
                Título
            </th>
            <th width="70%">
                Descripción
            </th>
        </tr>
        <tbody id="tablaBusqueda">
            </tbody>
        </table>
    </div>

```

```
</table>
```

```
</div>
```

```
<!-- div entrega de ejercicios -->
```

```
<div id="divEntregarEjercicio">
```

```
<form style="border-style: inset;">
```

```
<label for="slcEjerciciosPendientes" style="font-size: 13px;">Seleccionar <br> ejercicio</label>
```

```
<select id="slcEjerciciosPendientes">
```

```
</select>
```

```
<input type="button" value="Seleccionar" id="btnSeleccionarEjercicio">
```

```
</form>
```

```
<div id="verFormularioEntrega">
```

```
<p id="pEjercicio"></p>
```

```
<form style="border-style: inset;">
```

```
<label for="sonido">Adjuntar audio</label>
```

```
<input type="file" src="audios/" name="sonido" id="sonido">
```

```
<input type="button" value="Entregar" id="btnEnviarEntrega">
```

```
</form>
```

```
<p id="resultado">
```

```
</p>
```

```
</div>
```

```
</div>
```

```
<!-- div para ver ejercicios resueltos -->
```

```
<div id="divVerEjerciciosResueltos">
```

```
<h3>Listado de ejercicios resueltos</h3>
```

```
<form style="border-style: inset;">
```

```

<table border="1" style="width: 100%;">
  <tr>
    <th width="25%">
      Título
    </th>
    <th width="25%">
      Estado
    </th>
    <th width="50%">
      Fallo
    </th>
  </tr>

  <tbody id="tablaEjerResueltos">

  </tbody>

</table>
</form>
</div>

```

```

<!-- div para ver estadísticas del perfil tipo alumno -->
<div id="divVerEstadisticasDeAlumno">
  <p id="pEjerResueltosParaSuNivel"></p>
</div>

```

```

</div><!-- FIN CONTENIDO -->
<div id="pie">
  <a href="https://www.ort.edu.uy">Footer</a>

```

```
</div><!-- FIN PIE -->
</div><!-- FIN CONTENEDOR -->
<script src="codigo.js" type="text/javascript"></script>
</body>
</html>
```

---JS---

```
//-----\\
//-----VISUALIZACIONES-----\\
//-----\\
```

```
//creo funcion para cargar datos al comienzo al ejecutar a pagina
```

```
function load() {
```

```

    ocultarTodo();
    desplegable();
```

```
    document.getElementById("btnCrearDocente").style.display = "block";
    document.getElementById("btnCrearAlumno").style.display = "block";
    document.getElementById("btnIniciarSesion").style.display = "block";
```

```

}
}
```

```
//funcion del combo desplegable de docentes para seleccionar
```

```
function desplegable(){
    document.getElementById("tituloDocente").style.display = "block";
    document.querySelector("#slcprofe").innerHTML = "";
```



```
for (let i = 0; i < docentes.length; i++) {  
    document.querySelector("#slcprofe").innerHTML +=  
    `<option value="${docentes[i].nombreUsuario}">${docentes[i].nombre} (${docentes[i].nombreUsuario})</option>`  
}  
}
```

```
// funcion para ocultar todos los elementos de la página  
function ocultarTodo(){
```

```
    document.querySelector("#divCrearUsuario").style.display = "none";  
    document.querySelector("#divIniciarSesion").style.display = "none";  
    document.querySelector("#divAsignarNivel").style.display = "none";  
    document.querySelector("#divGenerarEjercicio").style.display = "none";  
    document.querySelector("#divBuscarEjercicio").style.display = "none";  
    document.querySelector("#divEntregarEjercicio").style.display = "none";  
    document.querySelector("#verFormularioEntrega").style.display = "none";  
    document.querySelector("#divDevoluciones").style.display = "none";  
    document.querySelector("#formularioDeCalificacion").style.display = "none";  
    document.querySelector("#divEstadisticasDeDocente").style.display = "none";  
    document.querySelector("#divVerEjerciciosResueltos").style.display = "none";  
    document.querySelector("#divVerEstadisticasDeAlumno").style.display = "none";
```

```
    document.querySelector("#btnCrearDocente").style.display = "none";  
    document.querySelector("#btnCrearAlumno").style.display = "none";  
    document.querySelector("#btnIniciarSesion").style.display = "none";  
    document.querySelector("#btnCerrarSesion").style.display = "none";
```

```
    document.querySelector("#btnAsignarNivel").style.display = "none";  
    document.querySelector("#btnPlantearEjercicio").style.display = "none";  
    document.querySelector("#btnDevoluciones").style.display = "none";  
    document.querySelector("#btnInformacionEstadistica").style.display = "none";
```

```
document.querySelector("#btnEjerciciosPendientes").style.display = "none";
document.querySelector("#btnRealizarEntrega").style.display = "none";
document.querySelector("#btnEjerciciosResueltos").style.display = "none";
document.querySelector("#btnInfoEstadistica").style.display = "none";
```

```
document.getElementById("tituloDocente").style.display = "none";
}
```

```
window.onload = load;
```

```
// solapa de visualización para crear perfil docente
document.querySelector("#btnCrearDocente").addEventListener("click", crearDocente);
function crearDocente(){
    ocultarTodo();
    document.querySelector("#slcprofe").style.display = "none";
    document.querySelector("#divCrearUsuario").style.display = "block";
    document.querySelector("#btnCrearAlumno").style.display = "block";
    document.querySelector("#btnIniciarSesion").style.display = "block";
    document.querySelector("#slcprofe").innerHTML = "";
    document.querySelector("#slcprofe").innerHTML =
        `<option value="no"></option>`
}
```

```
// solapa de visualización para crear perfil alumno
document.querySelector("#btnCrearAlumno").addEventListener("click", crearAlumno);
function crearAlumno(){
    ocultarTodo();
    document.querySelector("#slcprofe").style.display = "block";
    document.querySelector("#divCrearUsuario").style.display = "block";
    document.querySelector("#btnIniciarSesion").style.display = "block";
    document.querySelector("#btnCrearDocente").style.display = "block";
```

```
desplegable();
```

```
}
```

```
// solapa de visualización para iniciar sesión
```

```
document.querySelector("#btnIniciarSesion").addEventListener("click", iniciarSesion);
```

```
function iniciarSesion(){
```

```
    ocultarTodo();
```

```
    document.querySelector("#divIniciarSesion").style.display = "block";
```

```
    document.querySelector("#btnCrearDocente").style.display = "block";
```

```
    document.querySelector("#btnCrearAlumno").style.display = "block";
```

```
}
```

```
/*-----
```

```
//----- CLASES ----- \\
```

```
-----*/
```

```
class Docente{
```

```
    constructor(nombre,nombreUsuario,pass,alumnado){
```

```
        this.nombre = nombre;
```

```
        this.nombreUsuario = nombreUsuario;
```

```
        this.pass = pass;
```

```
        this.alumnado=alumnado = [];
```

```
    }
```

```
}
```

```
class Alumno{
```

```
    constructor(nombre,nombreUsuario,pass,docente,nivel){
```

```
        this.nombre = nombre;
```

```
    this.nombreUsuario = nombreUsuario;
    this.pass = pass;
    this.docente = docente;
    this.nivel = nivel;
}
}
```

```
class Ejercicio{
    constructor(autor, titulo, foto, descripcion, nivel){
        this.autor=autor;
        this.titulo=titulo;
        this.foto=foto;
        this.descripcion=descripcion;
        this.nivel=nivel;
    }
}
```

```
class Entrega{
    constructor(autor, titulo, audio, docente, nivel, estado,devolucion){
        this.autor=autor;
        this.titulo=titulo;
        this.audio=audio;
        this.docente=docente;
        this.nivel=nivel;
        this.estado=estado;
        this.devolucion=devolucion;
    }
}
```

```
//-----
```

```
/*-----  
// PRE-CARGA DE DATOS \\  
-----*/
```

```
let docentes = [];  
let alumnos = [];  
let ejercicios = [];  
let entregas = [];  
function preCargas(){  
  let d1 =new Docente("Jose", "jos45", "Asd4");
```

```
  let a1 =new Alumno("alum1", "alumno1", "Asd4", "jos45", "Inicial");  
  let a2 =new Alumno("alum2", "alumno2", "Asd4", "jos45", "Intermedio");  
  let a3 =new Alumno("alum8", "alumno8", "Asd4", "jos45", "Intermedio");  
  let a4 =new Alumno("alum9", "alumno9", "Asd4", "jos45", "Inicial");  
  let a5 =new Alumno("alum11", "alumno11", "Asd4", "jos45", "Avanzado");  
  let a6 =new Alumno("alum28", "alumno28", "Asd4", "jos45", "Intermedio");  
  let a7 =new Alumno("alum29", "alumno29", "Asd4", "jos45", "Intermedio");  
}
```

```
let ej1 =new Ejercicio("jos45", "E mayor", {name: "e_mayor.png"}, "Es una escala mayor basada en E, su firma de clave tiene  
cuatro objetos punzantes . Su relativa menor es Do menor sostenida", "Inicial");  
let ej2 =new Ejercicio("jos45", "E minor", {name: "a_mayor.jpg"}, "Es una escala mayor basada en E, disinta a mayor, como vi  
mos en clase", "Inicial");  
let ej3 =new Ejercicio("jos45", "E mayor - intermedio", {name: "e_mayor.png"}, "Enviar dicha escala pero en un tempo de 120  
bpm", "Intermedio");  
let ej4 =new Ejercicio("jos45", "E mayor - pre avanzado", {name: "e_mayor.png"}, "Realizar dicha escala pero en un tempo de  
140 bpm", "Intermedio");  
let ej5 =new Ejercicio("jos45", "Combinado en Do", {name: "e_mayor.png"}, "Realizar dicha escala pero en combinación con Do"  
, "Intermedio");  
let ej6 =new Ejercicio("jos45", "Old McDonald", {name: "ej7.png"}, "Tocarla sin exigencias de tempo", "Intermedio");  
let ej7 =new Ejercicio("jos45", "La marcha de los santos", {name: "ej8.png"}, "Tocarla sin exigencias de tempo", "Intermedio  
");
```

```
let ej8 =new Ejercicio("jos45", "E major con variaciones", {name: "e_major.png"}, "Realizar dicha escala en un tempo de 140 bpm y luego en Do también a 140", "Avanzado");
let ej9 =new Ejercicio("jos45", "La marcha de los santos", {name: "ej8.png"}, "Tocarla en un tempo de 120 bpm", "Avanzado");

let en1 =new Entrega("alumno1", "E major", {name: "E_Major.mp3"}, "jos45", "Inicial", "Enviado para calificar", "");
let en2 =new Entrega("alumno1", "E minor", {name: "E_Major.mp3"}, "jos45", "Inicial", "Calificado", "6. Procure afinar bien la próxima");
let en3 =new Entrega("alumno2", "E major - intermedio", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en4 =new Entrega("alumno2", "E major - pre avanzado", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en5 =new Entrega("alumno8", "E major - intermedio", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en6 =new Entrega("alumno8", "E major - pre avanzado", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en7 =new Entrega("alumno8", "Combinado en Do", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en8 =new Entrega("alumno8", "Old McDonald", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en9 =new Entrega("alumno9", "E major", {name: "E_Major.mp3"}, "jos45", "Inicial", "Enviado para calificar", "");
let en10 =new Entrega("alumno11", "E major con variaciones", {name: "E_Major.mp3"}, "jos45", "Avanzado", "Enviado para calificar", "");
let en11 =new Entrega("alumno28", "E major - intermedio", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en12 =new Entrega("alumno28", "E major - pre avanzado", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en13 =new Entrega("alumno28", "Combinado en Do", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en14 =new Entrega("alumno28", "Old McDonald", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en15 =new Entrega("alumno29", "E major - intermedio", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
let en16 =new Entrega("alumno29", "E major - pre avanzado", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar", "");
```

```
let en17 =new Entrega("alumno29", "Combinado en Do", {name: "E_Major.mp3"}, "jos45", "Intermedio", "Enviado para calificar",
"");
```

```
//-----
```

```
let d2 =new Docente("Maria", "mar99", "Asd4");
```

```
let a8 =new Alumno("alum5", "alumno5", "Asd4", "mar99", "Avanzado");
let a9 =new Alumno("alum6", "alumno6", "Asd4", "mar99", "Inicial");
let a10 =new Alumno("alum14", "alumno14", "Asd4", "mar99", "Inicial");
let a11 =new Alumno("alum15", "alumno15", "Asd4", "mar99", "Avanzado");
let a12 =new Alumno("alum16", "alumno16", "Asd4", "mar99", "Intermedio");
let a13 =new Alumno("alum18", "alumno18", "Asd4", "mar99", "Intermedio");
let a14 =new Alumno("alum19", "alumno19", "Asd4", "mar99", "Avanzado");
let a15 =new Alumno("alum20", "alumno20", "Asd4", "mar99", "Avanzado");
let a16 =new Alumno("alum22", "alumno22", "Asd4", "mar99", "Avanzado");
```

```
alumD2 = {nombre: "alum5",alumno: "alumno5",cantidadEjerResueltos: 3},{nombre: "alum6",alumno: "alumno6",
,cantidadEjerResueltos: 1},{nombre: "alum14",alumno: "alumno14",cantidadEjerResueltos: 2},{nombre: "alum15",alumno: "alumno15",
,cantidadEjerResueltos: 3},{nombre: "alum16",alumno: "alumno16",cantidadEjerResueltos: 1},{nombre: "alum18",alumno: "alumno18",
,cantidadEjerResueltos: 0},{nombre: "alum19",alumno: "alumno19",cantidadEjerResueltos: 1},{nombre: "alum20",alumno: "alumno20",
,cantidadEjerResueltos: 2},{nombre: "alum22",alumno: "alumno22",cantidadEjerResueltos: 4};
```

```
let ej10 =new Ejercicio("mar99", "E major", {name: "e_major.png"}, "Es una escala mayor basada en E, su firma de clave tiene
cuatro objetos punzantes . Su relativa menor es Do menor sostenida", "Inicial");
```

```
let ej11 =new Ejercicio("mar99", "Escala cromatica",{name: "a_major.jpg"}, "La escala cromática es aquella que creamos al to
car las doce notas de nuestro sistema musical", "Inicial");
```

```
let ej12 =new Ejercicio("mar99", "E minor", {name: "a_major.jpg"}, "Es una escala mayor basada en E, disinta a mayor, como v
imos en clase", "Inicial");
```

```
let ej13 =new Ejercicio("mar99", "Solo: cryin -Aerosmith-", {name: "a_major.jpg"}, "A, A#, A#5, C, C#m, C5 o D", "Avanzado
");
```

```
let ej14 =new Ejercicio("mar99", "Solo: november rain -Guns and Roses-", {name: "a_major.jpg"}, "A, Bm, C, C#m, D, E, F, F#m
o G", "Avanzado");
```

```
let ej15 =new Ejercicio("mar99", "Ejercicio 5", {name: "a_major.jpg"}, "Heavy metal Mi menor o Em", "Avanzado");
let ej16 =new Ejercicio("mar99", "Ejercicio 8", {name: "a_major.jpg"}, "Yngwie Malmsteen arpeggios en La menor o Am, Ahora ve
remos ejercicios por estilos de guitarristas", "Avanzado");
let ej17 =new Ejercicio("mar99", "Ejercicio 11", {name: "a_major.jpg"}, "Yngwie Extracto del tema -Trilogy Suite Opus 5-", "
Avanzado");
let ej18 =new Ejercicio("mar99", "E major - intermedio", {name: "e_major.png"}, "Enviar dicha escala pero en un tempo de 120
bpm", "Intermedio");

let en18 =new Entrega("alumno5", "Solo: cryin -Aerosmith-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para califi
car", "");
let en19 =new Entrega("alumno5", "Solo: november rain -Guns and Roses-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviad
o para calificar", "");
let en20 =new Entrega("alumno5", "Ejercicio 5", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calificar", "");
let en21 =new Entrega("alumno6", "E major", {name: "E_Major.mp3"}, "mar99", "Inicial", "Enviado para calificar", "");
let en22 =new Entrega("alumno14", "E major", {name: "E_Major.mp3"}, "mar99", "Inicial", "Enviado para calificar", "");
let en23 =new Entrega("alumno14", "Escala cromatica", {name: "E_Major.mp3"}, "mar99", "Inicial", "Enviado para calificar", "
");
let en24 =new Entrega("alumno15", "Solo: cryin -Aerosmith-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calif
icar", "");
let en25 =new Entrega("alumno15", "Solo: november rain -Guns and Roses-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Envia
do para calificar", "");
let en26 =new Entrega("alumno15", "Ejercicio 5", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calificar", "");
let en27 =new Entrega("alumno16", "E major - intermedio", {name: "E_Major.mp3"}, "mar99", "Intermedio", "Enviado para califi
car", "");
let en28 =new Entrega("alumno19", "Solo: cryin -Aerosmith-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calif
icar", "");
let en29 =new Entrega("alumno20", "Solo: cryin -Aerosmith-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calif
icar", "");
let en30 =new Entrega("alumno20", "Solo: november rain -Guns and Roses-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Envia
do para calificar", "");
let en31 =new Entrega("alumno22", "Solo: cryin -Aerosmith-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calif
icar", "");
let en32 =new Entrega("alumno22", "Solo: november rain -Guns and Roses-", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Envia
```



```

do para calificar", "");
let en33 =new Entrega("alumno22", "Ejercicio 5", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calificar", "");
let en34 =new Entrega("alumno22", "Ejercicio 8", {name: "E_Major.mp3"}, "mar99", "Avanzado", "Enviado para calificar", "");

//-----

let d3 =new Docente("Nelson", "Nel20", "Asd4");

let a17 =new Alumno("alum3", "alumno3", "Asd4", "Nel20", "Inicial");
let a18 =new Alumno("alum4", "alumno4", "Asd4", "Nel20", "Inicial");
let a19 =new Alumno("alum7", "alumno7", "Asd4", "Nel20", "Intermedio");
let a20 =new Alumno("alum10", "alumno10", "Asd4", "Nel20", "Avanzado");
let a21 =new Alumno("alum23", "alumno23", "Asd4", "Nel20", "Inicial");
let a22 =new Alumno("alum12", "alumno12", "Asd4", "Nel20", "Inicial");
let a23 =new Alumno("alum13", "alumno13", "Asd4", "Nel20", "Inicial");

alumD3 = {nombre: "alum3",alumno: "alumno3",cantidadEjerResueltos: 2},{nombre: "alum4",alumno: "alumno4",cantidadEjerResueltos: 2},{nombre: "alum7",alumno: "alumno7",cantidadEjerResueltos: 0},{nombre: "alum10",alumno: "alumno10",cantidadEjerResueltos: 0},{nombre: "alum23",alumno: "alumno23",cantidadEjerResueltos: 1},{nombre: "alum12",alumno: "alumno12",cantidadEjerResueltos: 3},{nombre: "alum13",alumno: "alumno13",cantidadEjerResueltos: 0};

let ej19 =new Ejercicio("Nel20", "E mayor", {name: "e_mayor.png"}, "Es una escala mayor basada en E, su firma de clave tiene cuatro objetos punzantes . Su relativa menor es Do menor sostenida", "Inicial");
let ej20 =new Ejercicio("Nel20", "E menor", {name: "a_mayor.jpg"}, "Es una escala mayor basada en E, disinta a mayor, como vimos en clase", "Inicial");
let ej21 =new Ejercicio("Nel20", "Ejercicio de guitarra 1", {name: "a_mayor.jpg"}, "Mi recomendación sería por ejemplo que toques el ejercicio de la 1ª y 2ª cuerda en canciones que SOLO utilizan notas de esas cuerdas", "Inicial");
let ej22 =new Ejercicio("Nel20", "Ejercicio de guitarra 2", {name: "a_mayor.jpg"}, "En este ejercicio de guitarra introducimos la tercera cuerda. Esta cuerda sólo se ha de tocar si se conocen las 6 notas básicas de las cuerdas 1 y 2", "Inicial");
let ej23 =new Ejercicio("Nel20", "Escala mixolidia de sol", {name: "a_mayor.jpg"}, "Esta escala contiene las 8 notas que se han visto antes. Tocala ascendente y descendentemente.", "Inicial");
let ej24 =new Ejercicio("Nel20", "Arpeggio 1", {name: "a_mayor.jpg"}, "Estudialos con metronomo y lentamente. A medida que te encuentres comodo, aumenta progresivamente la velocidad.", "Intermedio");

```

```
let ej25 =new Ejercicio("Nel20", "Ejercicio avanzado 1", {name: "a_mayor.jpg"}, "Ejercicios basados en el estilo de Steve Vai. Vai utiliza un sin numero de tecnicas juntas como pull off, slide up y down etc", "Avanzado");
```

```
let en35 =new Entrega("alumno3", "E mayor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en36 =new Entrega("alumno3", "E menor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en37 =new Entrega("alumno4", "E mayor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en38 =new Entrega("alumno4", "E menor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en39 =new Entrega("alumno23", "E mayor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en40 =new Entrega("alumno12", "E mayor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en41 =new Entrega("alumno12", "E menor", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");  
let en42 =new Entrega("alumno12", "Escala mixolidia de sol", {name: "E_Mayor.mp3"}, "Nel20", "Inicial", "Enviado para calificar", "");
```

```
//pusheo docentes  
docentes.push(d1,d2,d3);  
//pusheo alumnos  
alumnos.push(a1 , a2 , a3 , a4 , a5 , a6 , a7 , a8 , a9 , a10 , a11 , a12 , a13 , a14 , a15 , a16 , a17 , a18 , a19 , a20 , a21 , a22 , a23);  
//pusheo ejercicios  
ejercicios.push(ej1 , ej2 , ej3 , ej4 , ej5 , ej6 , ej7 , ej8 , ej9 , ej10 , ej11 , ej12 , ej13 , ej14 , ej15 , ej16 , ej17 , ej18 , ej19 , ej20 , ej21 , ej22 , ej23 , ej24 , ej25);  
//pusheo entregas  
entregas.push(en1 , en2 , en3 , en4 , en5 , en6 , en7 , en8 , en9 , en10 , en11 , en12 , en13 , en14 , en15 , en16 , en17 , en18 , en19 , en20 , en21 , en22 , en23 , en24 , en25 , en26 , en27 , en28 , en29 , en30 , en31 , en32 , en33 , en34 , en35 , en36 , en37 , en38 , en39 , en40 , en41 , en42);
```

```
//pusheo el alumnado del primer docente  
docentes[0].alumnado.push({nombre: "alum1",alumno: "alumno1",cantidadEjerResueltos: 2},{nombre: "alum2",alumno: "alumno2",cantidadEjerResueltos: 2},{nombre: "alum8",alumno: "alumno8",cantidadEjerResueltos: 4},{nombre: "alum9",alumno: "alumno9",cantidadEjerResueltos: 1},{nombre: "alum11",alumno: "alumno11",cantidadEjerResueltos: 1},{nombre: "alum28",alumno: "alumno28",cantidadEjerResueltos: 4},{nombre: "alum29",alumno: "alumno29",cantidadEjerResueltos: 3});  
//pusheo el alumnado del segundo docente
```

```

docentes[1].alumnado.push({nombre: "alum5",alumno: "alumno5",cantidadEjerResueltos: 3},{nombre: "alum6",alumno: "alumno6",
,cantidadEjerResueltos: 1},{nombre: "alum14",alumno: "alumno14",cantidadEjerResueltos: 2},{nombre: "alum15",alumno: "alumno15",
,cantidadEjerResueltos: 3},{nombre: "alum16",alumno: "alumno16",cantidadEjerResueltos: 1},{nombre: "alum18",alumno: "alumno18",
,cantidadEjerResueltos: 0},{nombre: "alum19",alumno: "alumno19",cantidadEjerResueltos: 1},{nombre: "alum20",alumno: "alumno20",
,cantidadEjerResueltos: 2},{nombre: "alum22",alumno: "alumno22",cantidadEjerResueltos: 4}});
//pusheo el alumnado del tercer docente
docentes[2].alumnado.push({nombre: "alum3",alumno: "alumno3",cantidadEjerResueltos: 2},{nombre: "alum4",alumno: "alumno4",
,cantidadEjerResueltos: 2},{nombre: "alum7",alumno: "alumno7",cantidadEjerResueltos: 0},{nombre: "alum10",alumno: "alumno10",
,cantidadEjerResueltos: 0},{nombre: "alum23",alumno: "alumno23",cantidadEjerResueltos: 1},{nombre: "alum12",alumno: "alumno12",
,cantidadEjerResueltos: 3},{nombre: "alum13",alumno: "alumno13",cantidadEjerResueltos: 0}});
}
preCargas();

```

```

//-----\\
//-----REGISTRO DE USUARIOS-----\\
//-----\\

document.querySelector("#btnRegistrar").addEventListener("click", registrar);
function registrar(){
    let nombre = document.querySelector("#txtNombre").value.trim();
    let nombreUsuario = document.querySelector("#txtUsuario").value.trim();

```

```
let pass = document.querySelector("#txtPass").value.trim();
let profe = document.querySelector("#slcprofe").value;
let nivel = "Inicial";

let control = controles(nombreUsuario,pass);

if (control){

    if(profe === "no"){
        docentes.push(new Docente(nombre, nombreUsuario, pass, ""));
        alert("Usuario tipo docente creado con éxito");
    }

    document.querySelector("#txtNombre").value = "";
    document.querySelector("#txtUsuario").value = "";
    document.querySelector("#txtPass").value = "";
    return;
}

alumnos.push(new Alumno(nombre, nombreUsuario, pass, profe, nivel));

for (let i = 0; i < docentes.length; i++) {
    if(profe === docentes[i].nombreUsuario ){
        docentes[i].alumnado.push({nombre: nombre , alumno: nombreUsuario, cantidadEjerResueltos: 0})
    }
}

alert("Usuario tipo alumno creado con éxito");

}
```

```
document.querySelector("#txtNombre").value = "";
document.querySelector("#txtUsuario").value = "";
document.querySelector("#txtPass").value = "";
}
```

```
// verificacion de requisitos de nombre de usuario y contraseña
function controles(nombreUsuario,pass){
```

```
    nombreUsuario = nombreUsuario.toLowerCase();
    // control (1) verificar si existe usuario de tipo docente o alumno con el mismo nombre
    let existeUsuarioDocente = false;
    for (let i = 0; i < docentes.length; i++) {
        if(nombreUsuario === docentes[i].nombreUsuario.toLowerCase()){
            existeUsuarioDocente = true;
            alert("Error, este usuario de docente ya existe");
        }
    }
}
```

```
let existeUsuarioAlumno = false;
for (let i = 0; i < alumnos.length; i++) {
    if(nombreUsuario === alumnos[i].nombreUsuario.toLowerCase()){
        existeUsuarioAlumno = true;
    }
}
```

```
    alert("Error, este usuario de alumno ya existe");  
  }  
  }  
}
```

```
//control (2) que la contraseña no tenga más de 4 caracteres  
let menorDeCuatro = false;
```

```
if(pass.length < 4 ){  
  menorDeCuatro = true;  
  alert("Error, la contraseña debe tener un mínimo de 4 caracteres");  
}
```

```
//control (3) que la contraseña contenga al menos una minúscula una mayúscula y un número
```

```
let tieneMayus = false;  
let tieneMinus = false;  
let tieneNumeros = false;  
for(let i = 0; i < pass.length;i++){  
  let caracter = pass.charAt(i);  
  //verifico si es mayuscula o minuscula (y no un numero)  
  if(isNaN(caracter) && caracter === caracter.toUpperCase()){  
    tieneMayus = true;  
  }  
  if(isNaN(caracter) && caracter === caracter.toLowerCase()){  
    tieneMinus = true;  
  }  
  //verifico si es un numero  
  if(!isNaN(caracter)){  
    tieneNumeros = true;  
  }  
}
```

```
if(tieneMinus == false || tieneMayus == false || tieneNumeros == false){  
    alert("Error, la contraseña debe tener al menos una mayúscula, una minúscula y un número");  
}
```

```
if(existeUsuarioDocente == false && existeUsuarioAlumno == false && menorDeCuatro == false && tieneMayus == true  
&& tieneMinus == true && tieneNumeros == true ){  
    return true;  
}
```

```
return false;  
}
```

```
//-----\\  
//-----LOGUEO DE USUARIOS-----\\  
//-----\\
```

```
let usuarioLogueado = []; // alojo al usuario logueado en un array para comparar datos con él  
document.querySelector("#btnLogueo").addEventListener("click", loguearse);  
function loguearse(){
```

```

let usuario = document.querySelector("#txtuser").value.trim().toLowerCase();
let usuarioPass = document.querySelector("#txtcontra").value.trim();

let flag = false; // compruebo por bandera que el usuario que se intenta logear exista y coincida usuario y contraseña
for(let i=0 ; i < docentes.length; i++){

    if(docentes[i].nombreUsuario.toLowerCase() === usuario && docentes[i].pass===usuarioPass)
    {
        flag = true;
        let usuarioIngresado = docentes[i];
        usuarioLogueado.push(usuarioIngresado);

        alert("usuario " + usuarioIngresado.nombreUsuario+ " logueado");
        console.log(usuarioLogueado[0]);

        ocultarTodo();
        document.querySelector("#btnCerrarSesion").style.display = "block";
        document.querySelector("#btnAsignarNivel").style.display = "block";
        document.querySelector("#btnPlantearEjercicio").style.display = "block";
        document.querySelector("#btnDevoluciones").style.display = "block";
        document.querySelector("#btnInformacionEstadistica").style.display = "block";

    }
}

for(let i=0 ; i < alumnos.length; i++){

    if(alumnos[i].nombreUsuario.toLowerCase() === usuario && alumnos[i].pass===usuarioPass)
    {
        flag = true;
    }
}

```



```
    let usuarioIngresado = alumnos[i];
    usuarioLogueado.push(usuarioIngresado);
    alert("usuario " + usuarioIngresado.nombreUsuario + " logueado");
    console.log(usuarioLogueado[0]);
    ocultarTodo();
    document.querySelector("#btnCerrarSesion").style.display = "block";
    document.querySelector("#btnEjerciciosPendientes").style.display = "block";
    document.querySelector("#btnRealizarEntrega").style.display = "block";
    document.querySelector("#btnEjerciciosResueltos").style.display = "block";
    document.querySelector("#btnInfoEstadistica").style.display = "block";
}
}
```

```
if(flag == false){
    alert("Usuario o contraseña incorrecta");
}
```

```
//Blanqueo los cuadros de texto
document.querySelector("#txtuser").value = "";
document.querySelector("#txtcontra").value = "";
```

```
}
```

```
//cierre de sesión
document.querySelector("#btnCerrarSesion").addEventListener("click", cerrarSesion);
function cerrarSesion(){
    usuarioLogueado.splice(0,1);
    console.log(usuarioLogueado[0]);
}
```

```
load();
```

```
}
```

```
/*-----
```

```
// FUNCIONALIDADES DE USUARIO TIPO DOCENTE //
```

```
-----*/
```

```
//----- asignar nivel a alumnos-----\\
```

```
document.querySelector("#btnAsignarNivel").addEventListener("click", asignarNivel);
```

```
function asignarNivel(){
```

```
document.querySelector("#formularioDeCalificacion").style.display = "none";
```

```
document.querySelector("#divDevoluciones").style.display = "none";
```

```
document.querySelector("#divEstadisticasDeDocente").style.display = "none";
```

```
document.querySelector("#slcAscenderAlumno").innerHTML = "";
```

```
document.querySelector("#tablaNiveles").innerHTML = "";
```

```
for (let i = 0; i < alumnos.length; i++) { //listo alumnos en un combo desplegable para que pueda seleccionar el alumno a ascender
```

```
if(usuarioLogueado[0].nombreUsuario === alumnos[i].docente){
```

```
document.querySelector("#slcAscenderAlumno").innerHTML +=
```

```
`<option value="${alumnos[i].nombreUsuario}">${alumnos[i].nombre} (${alumnos[i].nombreUsuario})</option>`;
```

```
//genero una tabla de referencia para ver todos los alumnos y sus niveles en tiempo real
document.querySelector("#tablaNiveles").innerHTML +=
`<tr>
<td>${alumnos[i].nombre}</td>
<td>${alumnos[i].nombreUsuario}</td>
<td>${alumnos[i].nivel}</td>
</tr>`

}
}
```

```
document.querySelector("#divGenerarEjercicio").style.display = "none";
document.querySelector("#divAsignarNivel").style.display = "block";
```

```
}
```

```
document.querySelector("#btnAscensoAlumno").addEventListener("click", ascenso);
function ascenso(){
```

```
let alumnoAscenso = document.querySelector("#slcAscenderAlumno").value;
```

```
for (let i = 0; i < alumnos.length; i++) {
```

```
if(alumnoAscenso === alumnos[i].nombreUsuario){
```

```
if(alumnos[i].nivel === "Inicial"){
```

```
alumnos[i].nivel = "Intermedio";
```

```
alert("El alumno "+alumnos[i].nombreUsuario+" ha alcanzado el nivel intermedio");
```

```
asignarNivel();
```

```
return;
```

```
}
```



```
document.querySelector("#btnCrearEjercicio").addEventListener("click", crearEjercicio);
function crearEjercicio(){
```

```
    let titulo = document.querySelector("#txtTituloEjercicio").value;
    let foto = document.querySelector("#foto").files[0];
    let descripcion = document.querySelector("#txtDescripcion").value;
    let dificultad = document.querySelector("#slcNivelDeEjercicio").value;
```

```
    let controles = controlEjercicio(titulo, descripcion, foto);
```

```
    if(controles){
        ejercicios.push(new Ejercicio(usuarioLogueado[0].nombreUsuario, titulo, foto, descripcion, dificultad));
        alert("Ejercicio creado con éxito");
    }
}
```

```
document.querySelector("#txtTituloEjercicio").value = "";
document.querySelector("#txtDescripcion").value = "";
document.querySelector("#foto").value = "";
```

```
}
```

```
// verifico que los parámetros cumplan con lo requerido
```

```
function controlEjercicio(titulo, descripcion, foto){
    let sumaLength = titulo.length + descripcion.length;
    if(sumaLength > 200 || sumaLength < 20 || titulo.length === 0 || descripcion.length === 0 || foto === undefined ){
        alert("Error, los campos no pueden estar vacíos y debe contener mínimo 20 caracteres y un máximo de 200");
        return false;
    }
    return true;
}
```

```
//----- creo función para generar devoluciones de ejercicios-----\\
```

```
document.querySelector("#btnDevoluciones").addEventListener("click", ejerciciosParaCalificar);
```

```
function ejerciciosParaCalificar(){
```

```
    document.querySelector("#divAsignarNivel").style.display = "none";
```

```
    document.querySelector("#divGenerarEjercicio").style.display = "none";
```

```
    document.querySelector("#slcDevoluciones").innerHTML = "";
```

```
    document.querySelector("#divEstadisticasDeDocente").style.display = "none";
```

```
    document.querySelector("#divDevoluciones").style.display = "block";
```

```
    for (let i = 0; i < entregas.length; i++) {
```

```
        if(usuarioLogueado[0].nombreUsuario === entregas[i].docente){
```

```
            document.querySelector("#slcDevoluciones").innerHTML +=
```

```
                `<option value="${i}">${entregas[i].titulo}(${entregas[i].autor})</option>`;
```

```
        }
```

```
    }
```

```
}
```

```
document.querySelector("#btnSeleccionarEntrega").addEventListener("click", fSeleccionEntrega);
```

```
let posicionEntrega; //creo variable auxiliar para alojar posicion de la entrega en array para utilizarla en la devolucion
```

```
function fSeleccionEntrega(){
```

```
    let entregaSeleccionada = document.querySelector("#slcDevoluciones").value;
```

```
    // verifico que ya no se haya dado una devolucion previa, en caso que si me impide continuar
```

```
    if(entregas[entregaSeleccionada].estado === "Calificado"){
```

```
        alert("Este ejercicio ya fue calificado");
```

```
        return;
```

```

    }
}

document.querySelector("#formularioDeCalificacion").style.display = "block";

document.querySelector("#pEntrega").innerHTML =
`<div>
  <form style="border-style: inset;">
    <b>${entregas[entregaSeleccionada].titulo}</b>
    <p>Entregado por: ${entregas[entregaSeleccionada].autor}</p>
    <audio controls>
      <source src="audios/${entregas[entregaSeleccionada].audio.name}" type="audio/mpeg">
    </audio>
  </form>
</div> `
posicionEntrega = entregaSeleccionada;
}

document.querySelector("#btnEnviarDevolucion").addEventListener("click", envioDeDevolucion);
function envioDeDevolucion(){
  let textoDevolucion = document.querySelector("#txtDevolucion").value;
  if(textoDevolucion.length === 0){ alert("Error, campo vacío"); return; }
  entregas[posicionEntrega].devolucion = textoDevolucion;
  entregas[posicionEntrega].estado = "Calificado";
  document.querySelector("#txtDevolucion").value = "";
  document.querySelector("#formularioDeCalificacion").style.display = "none";
}
}

```

```

/*-----
// FUNCIONALIDAD DEL PERFIL TIPO ALUMNO \

```

```

-----*/
//----- Búsqueda de ejercicios----- \\
document.querySelector("#btnEjerciciosPendientes").addEventListener("click", realizarBusqueda);
function realizarBusqueda(){
    document.querySelector("#divVerEjerciciosResueltos").style.display = "none";
    document.querySelector("#divVerEstadisticasDeAlumno").style.display = "none";
    document.querySelector("#divEntregarEjercicio").style.display = "none";
    document.querySelector("#tablaBusqueda").innerHTML = "";
    document.querySelector("#divBuscarEjercicio").style.display = "block";
}

```

```

document.querySelector("#btnBusquedaDeEjercicio").addEventListener("click", buscarEjercicios);
function buscarEjercicios(){

```

```

    document.querySelector("#tablaBusqueda").innerHTML = "";

```

```

    let palabraBuscada = document.querySelector("#txtBuscarEjercicio").value.toLowerCase();
    if(palabraBuscada.length === 0){
        alert("Error, el campo no puede estar vacío");
        return;
    }
    let bandera = false;
    let coincidencias = 0; /*si encuentra la subcadena suma 1. Si no encuentra coincidencias queda en 0 y arroja
    el mensaje No hay resultados que coincidan con su búsqueda al final*/
    for (let i = 0; i < ejercicios.length; i++) {
        if(ejercicios[i].autor === usuarioLogueado[0].docente && ejercicios[i].nivel === usuarioLogueado[0].nivel &&
        ejercicios[i].titulo.toLowerCase().includes(palabraBuscada)){
            bandera = true;
            document.querySelector("#tablaBusqueda").innerHTML +=
            `<tr>
            <td>${ejercicios[i].titulo}</td>

```



```
    <td>${ejercicios[i].descripcion.slice(0,40)+"..."}</td>
  </tr>`
  coincidencias += 1;
}
```

```
if(ejercicios[i].autor === usuarioLogueado[0].docente && ejercicios[i].nivel === usuarioLogueado[0].nivel &&
  ejercicios[i].descripcion.toLowerCase().includes(palabraBuscada) && bandera == false){
  bandera = true;
  document.querySelector("#tablaBusqueda").innerHTML +=
  `<tr>
    <td>${ejercicios[i].titulo}</td>
    <td>${ejercicios[i].descripcion.slice(0,40)+"..."}</td>
  </tr>`
  bandera = false;
  coincidencias += 1;
}

}

document.querySelector("#txtBuscarEjercicio").value = "";
if(bandera == true){
  return;
}
if(coincidencias === 0){
  alert("No hay resultados que coincidan con su búsqueda");
}
}
```

```
// -----Realizar la entrega de un ejercicio----- \\

document.querySelector("#btnRealizarEntrega").addEventListener("click", verHacerEntrega);

let entregaAutor, entregaTitulo, entregaDocente, entregaEstado, entregaNivel; //variables auxiliares para el pusheo de entrega
s

function verHacerEntrega(){
    document.querySelector("#divBuscarEjercicio").style.display = "none";
    document.querySelector("#divVerEstadisticasDeAlumno").style.display = "none";
    document.querySelector("#divVerEjerciciosResueltos").style.display = "none";
    document.querySelector("#slcEjerciciosPendientes").innerHTML = "";
    document.querySelector("#divEntregarEjercicio").style.display = "block";

    for (let i = 0; i < ejercicios.length; i++) {
        if(ejercicios[i].autor === usuarioLogueado[0].docente && ejercicios[i].nivel === usuarioLogueado[0].nivel){
            document.querySelector("#slcEjerciciosPendientes").innerHTML +=
            `<option value="${ejercicios[i].titulo}">${ejercicios[i].titulo}</option>`;
        }
    }
}

document.querySelector("#btnSeleccionarEjercicio").addEventListener("click", verFormularioDeEntrega);
function verFormularioDeEntrega(){
    let ejercicioSeleccionado = document.querySelector("#slcEjerciciosPendientes").value;

    // verifico que el ejercicio no se haya entregado, en caso que si me impide continuar
    for (let i = 0; i < entregas.length; i++) {
        if(entregas[i].autor === usuarioLogueado[0].nombreUsuario && entregas[i].titulo === ejercicioSeleccionado){
```

```
    alert("Este ejercicio ya fue enviado para calificar");
    return;
}
}
```

```
for (let i = 0; i < ejercicios.length; i++) {
    if(ejercicios[i].autor === usuarioLogueado[0].docente && ejercicios[i].nivel === usuarioLogueado[0].nivel &&
        ejercicioSeleccionado === ejercicios[i].titulo ){
        document.querySelector("#pEjercicio").innerHTML =
        `<div>
        <form style="border-style: inset;">
        <h3>${ejercicios[i].titulo}</h3> <br><hr><br>  <br><hr><br> <p>${
ejercicios[i].descripcion}</p>
        </form>
        </div> `
        entregaAutor = usuarioLogueado[0].nombreUsuario;
        entregaTitulo = ejercicios[i].titulo;
        entregaDocente = ejercicios[i].autor;
        entregaNivel = ejercicios[i].nivel;
    }
}
```

```
document.querySelector("#verFormularioEntrega").style.display = "block";
}
```

```
document.querySelector("#btnEnviarEntrega").addEventListener("click", envioDeEntrega);
```

```

let arrAlumnado = [];
function envioDeEntrega(){
  //verifico que la entrega no esté vacía
  let audio = document.querySelector("#sonido").files[0];
  if(audio == undefined){
    alert("Error, se debe adjuntar un archivo para la entrega");
    return;
  }
  entregas.push(new Entrega(entregaAutor, entregaTitulo, audio, entregaDocente, entregaNivel, "Enviado para calificar", ""));
}

```

```

// cada vez que se envíe una entrega se suma a la cantidad total de envios por alumno para su posterior uso
//en estadísticas de docente
for (let i = 0; i < docentes.length; i++) {
  if(docentes[i].nombreUsuario === usuarioLogueado[0].docente ){

```

```

    for (let x = 0; x < docentes[i].alumnado.length; x++) {
      if(docentes[i].alumnado[x].alumno === usuarioLogueado[0].nombreUsuario){
        docentes[i].alumnado[x].cantidadEjerResueltos += 1;
        console.log(docentes[i].alumnado[x].cantidadEjerResueltos);
      }
    }

```

```

  }
}

```

```

}

```

```

}

```

```


```

```
document.querySelector("#sonido").value = "";
```

```
document.querySelector("#verFormularioEntrega").style.display = "none";
```

```
}]
```

```
//----- visualizar ejercicios resueltos-----\\
```

```
document.querySelector("#btnEjerciciosResueltos").addEventListener("click", visualizarEjerResueltos);
```

```
function visualizarEjerResueltos(){
```

```
document.querySelector("#divEntregarEjercicio").style.display = "none";
```

```
document.querySelector("#divBuscarEjercicio").style.display = "none";
```

```
document.querySelector("#divVerEstadisticasDeAlumno").style.display = "none";
```

```
document.querySelector("#tablaEjerResueltos").innerHTML = "";
```

```
document.querySelector("#divVerEjerciciosResueltos").style.display = "block";
```

```
}
```

```
for (let i = 0; i < entregas.length; i++){
```

```
if(entregas[i].autor === usuarioLogueado[0].nombreUsuario){
```

```
document.querySelector("#tablaEjerResueltos").innerHTML +=
```

```
`<td>${entregas[i].titulo}</td><td>${entregas[i].estado}</td><td>${entregas[i].devolucion}</td>`
```

```
}
```

```
}
```

```
}
```

```
/*-----  
-----// ESTADISTICAS DE USUARIOS \\-----  
-----*/
```

```
//----- informe de estadísticas ( docente ) -----\\  
document.querySelector("#btnInformacionEstadistica").addEventListener("click", informacionDocente);  
let todosEjerciciosEnviados = []; //genero un array local para alojar una lista de ejercicios para ese profe  
function informacionDocente(){  
    document.querySelector("#divAsignarNivel").style.display = "none";  
    document.querySelector("#divGenerarEjercicio").style.display = "none";  
    document.querySelector("#divDevoluciones").style.display = "none";  
    document.querySelector("#alumnoQueResolvioMas").innerHTML = "";  
    document.querySelector("#pCantidadEjerciciosPorNivel").innerHTML = "";  
    document.querySelector("#pCantidadEjerciciosResueltos").innerHTML = "";  
    document.querySelector("#slcAlumnosDeEseProfe").innerHTML = "";  
    document.querySelector("#divEstadisticasDeDocente").style.display = "block";  
}
```

```
  
//esto cuenta el total de los ejercicios realizados para cada profesor  
let contadorEntregas = 0;  
  
for (let i = 0; i < entregas.length; i++) {  
    if(entregas[i].docente === usuarioLogueado[0].nombreUsuario){  
        contadorEntregas++;  
    }  
}
```

```

    }
    usuarioLogueado[0].alumnado.sort(porNombre);
    // ordeno por criterio de mayor a menor cantidad de entregas por alumno
    function porNombre(a, b){
        if(a.cantidadEjerResueltos < b.cantidadEjerResueltos){
            return 1;
        }else if(b.cantidadEjerResueltos < a.cantidadEjerResueltos){
            return -1;
        }else{
            return 0;
        }
    }
}

//Muestro el nombre del alumno que ha resuelto más ejercicios, si hay alguno más que haya entregado la misma
//cantidad comparo la cantidad de ese alumno con las entregas realizadas en total y los muestro también
for (let i = 0; i < usuarioLogueado[0].alumnado.length; i++) {
    if(usuarioLogueado[0].alumnado[i].cantidadEjerResueltos === usuarioLogueado[0].alumnado[0].cantidadEjerResueltos){
        console.log(usuarioLogueado[0].alumnado[i]);
        document.querySelector("#alumnoQueResolvioMas").innerHTML +=
            `<td>${usuarioLogueado[0].alumnado[i].nombre}</td>
            <td>${usuarioLogueado[0].alumnado[i].alumno}</td>
            <td>${usuarioLogueado[0].alumnado[i].cantidadEjerResueltos}</td>`
    }
}

document.querySelector("#slcAlumnosDeEseProfe").innerHTML +=
    `<option value="${usuarioLogueado[0].alumnado[i].alumno}">${usuarioLogueado[0].alumnado[i].nombre}<${
usuarioLogueado[0].alumnado[i].alumno}</option>`;
}

```

```
document.querySelector("#pTotalEntregas").innerHTML = `  
<form style="border-style: inset;">  
  <p>Ejercicios que se han entregado en total: <b>${contadorEntregas}</b></p>  
</form> `;
```

```
document.querySelector("#divEstadisticasDeDocente").style.display = "block";  
}
```

```
document.querySelector("#btnDocEstadistica").addEventListener("click", totalesEstadisticosDocente);
```

```
function totalesEstadisticosDocente(){  
  let alumnoSeleccionado = document.querySelector("#slcAlumnosDeEseProfe").value;  
  let cantidadEjerciciosPorNivel = 0;  
  let cantidadEjerciciosResueltos = 0;
```

```
  for (let i = 0; i < alumnos.length; i++) {  
    if(alumnos[i].nombreUsuario === alumnoSeleccionado){  
      alumnoSeleccionado = alumnos[i];  
    }  
  }  
}
```

```
  for (let i = 0; i < ejercicios.length; i++) {  
    if(ejercicios[i].autor === usuarioLogueado[0].nombreUsuario && alumnoSeleccionado.nivel === ejercicios[i].nivel){  
      cantidadEjerciciosPorNivel += 1;  
    }  
  }  
}
```

```
  for (let i = 0; i < entregas.length; i++) {  
    
```



```
if(entregas[i].autor === alumnoSeleccionado.nombreUsuario){  
    cantidadEjerciciosResueltos += 1;  
}  
  
}
```

```
document.querySelector("#pCantidadEjerciciosPorNivel").innerHTML =  
    "La cantidad de ejercicios para el nivel "+alumnoSeleccionado.nivel+" es "+`<b>${cantidadEjerciciosPorNivel}</b>`;   
document.querySelector("#pCantidadEjerciciosResueltos").innerHTML =  
    "La cantidad de ejercicios resueltos por el alumno "+alumnoSeleccionado.nombreUsuario+" es "+`<b>${  
{cantidadEjerciciosResueltos}</b>`;   
  
}
```

```
//----- Estadísticas de perfil alumno-----\\  
document.querySelector("#btnInfoEstadistica").addEventListener("click", infoEstadisticaAlumno);  
function infoEstadisticaAlumno(){
```

```
document.querySelector("#divEntregarEjercicio").style.display = "none";  
document.querySelector("#divBuscarEjercicio").style.display = "none";  
document.querySelector("#divVerEjerciciosResueltos").style.display = "none";  
document.querySelector("#pEjerResueltosParaSuNivel").innerHTML = "";  
document.querySelector("#divVerEstadisticasDeAlumno").style.display = "block";  
  

```

```
let ejerTotalParaSuNivel = 0;  
let ejerResueltosParaSuNivel = 0;  
let ejerCalificados = 0;  
let ejerAunSinCalificar = 0;
```

```

for (let i = 0; i < ejercicios.length; i++) {
    if(ejercicios[i].nivel === usuarioLogueado[0].nivel && ejercicios[i].autor === usuarioLogueado[0].docente ){
        ejerTotalParaSuNivel++;
    }
}

// comparo el alumno logueado con la autoría de la entrega, su docente, su nivel y muestro los datos solicitados
for (let i = 0; i < entregas.length; i++) {
    if(entregas[i].autor === usuarioLogueado[0].nombreUsuario && entregas[i].nivel === usuarioLogueado[0].nivel ){
        ejerResueltosParaSuNivel++;
    }
    if(entregas[i].autor === usuarioLogueado[0].nombreUsuario && entregas[i].estado === "Calificado"){
        ejerCalificados++;
    }
    if(entregas[i].autor === usuarioLogueado[0].nombreUsuario && entregas[i].estado === "Enviado para calificar"){
        ejerAunSinCalificar++;
    }
}

let resultadoTotal = (ejerResueltosParaSuNivel * 100) / ejerTotalParaSuNivel;
if(isNaN(resultadoTotal)){ // para que en caso de que aún no hayan valores muestre 0 en lugar de NaN
    resultadoTotal = 0;
}
document.querySelector("#pEjerResueltosParaSuNivel").innerHTML =
`<form style="border-style: inset; width: 100%; margin: 0% 0% 0% 25%;">
<p style="padding-left: 5px;"> Ha resuelto el <b>${Math.round(resultadoTotal)}%</b> de las propuestas para su nivel </p>
</form>
<form style="border-style: inset; width: 100%; margin: 0% 0% 0% 25%;">
<p style="padding-left: 5px;"> <b>${ejerCalificados}</b> de las entregas han sido calificadas </p>
<p style="padding-left: 5px;"> <b>${ejerAunSinCalificar}</b> de las entregas están pendientes de calificación </p>

```

```
</form>`;
```

```
}
```