

## SEGUNDO DESEMPEÑO

### OBJETIVOS

- Utilizar las templates brindadas, inscrustando el PHP necesario para ser visualizado en el navegador. El diseño presentado no debería ser modificado, dado que solo se debe programar lo necesario para lograr la página dinámica.
- Usar los temas vistos hasta el momento.

### DESARROLLO

Vamos a trabajar en un **Panel de Administración**, solicitado por el área de Comunicaciones de nuestro cliente. Este panel permite a los usuarios ya registrados:

- Loguearse con **usuario** y clave válidos
- Registrar una **solicitud**: así le llamamos a un pedido que realiza un **usuario común**, para que otro usuario de **soporte técnico** o un **analista** pueda asistirlo.
- Listar las **solicitudes** que se han ido registrando, y dependiendo del tipo de solicitud, y el nivel del usuario (rol), podrán verse detalles.

Para estos procesos, la maqueta ya fue diseñada y preparada por el Área de Diseño. **No debe modificarse su estructura ni sus colores**, dado que esta área ya ha confirmado la disposición de las secciones y los colores del panel con el cliente.

Ahora nos queda a nosotros brindarle el dinamismo en sus datos. La original está disponible online<sup>1</sup>.



Nuestras interfaces se encuentran en formato *responsive*.

Tenemos diferentes pantallas en nuestro panel, deberás incluir el PHP necesario y guardar adecuadamente, la idea es que todas las estructuras del HTML ya listas se mantengan sin “romperse”.

Por favor leer atentamente todo este enunciado y luego comenzar a desarrollar.

Muchos éxitos! A comenzar!

<sup>1</sup> plantilla descargada desde <https://codecademy.com/pratiborsadiya/vali-admin/zip/master>

## LOGIN

*Panel de administración*

 **INGRESA AL PANEL**

---

Ingresar tus datos

Usuario (\*)

Clave (\*)

[Olvidaste la clave ?](#)


 **INGRESAR**

Aquí el usuario debe loguearse para ingresar al panel.

Los datos a ingresar son su usuario y clave, y tras presionar el botón se deberá validar para ingresar estas condiciones:

- el usuario **debe estar activo** para poder ingresar
- **usuario y clave correctos** (contra la base de datos)

*Panel de administración*

 **INGRESA AL PANEL**


---

Datos incorrectos.

Usuario (\*)

Clave (\*)

[Olvidaste la clave ?](#)

 **INGRESAR**

- si no se cumplen estas condiciones, se debe notificar como "Datos incorrectos". Notar que ya no se muestra el "Ingresar tus datos", sino que solo esta sección con el mensaje. Ver en la template está comentada esta parte.

- Solo si se cumplen las condiciones, se procede a **actualizar** para ese usuario la fecha actual de logueo en el campo "UltimaFechaAcceso" (lo detallamos más abajo).

- Luego a **abrir la sesión** de usuario correspondiente, alojando en ella los datos del usuario que necesites:

- Nombre , apellido, rol, imagen

y **redirigiendo** al index.

### Consideraciones para el proceso de Login

- Los **controles/elementos** del formulario deben ser del **tipo** adecuado y tener sus **propiedades** para que PHP logre recoger los datos ingresados en ellos.



Notar que esta plantilla trae el botón con el tag `<button></button>`. Deberás agregar el tipo y propiedades necesarias, para que funcione con el método de envío mas seguro.

- Los datos se enviarán de manera interna, sin mostrar valores por la barra de direcciones.
- En la **base de datos** debes almacenar por lo menos estos datos de cada **usuario**:
  - ✓ Un identificador único
  - ✓ Nombre y apellido (separados)
  - ✓ Foto o imagen (se alojan en la carpeta *images/team*)
  - ✓ Email (será el login)
  - ✓ Clave (encriptada con MD5) (La longitud del MD5 es de 32 caracteres), por tal este campo debería tener como mínimo esa longitud.
  - ✓ **Rol** (esto corresponde a una referencia numérica a otra tabla de Roles, debajo se detalla)
  - ✓ Última fecha de acceso (se almacena apenas se loguea), formato mysql.
  - ✓ Si está activo o no (un valor que indique esto, puede ser 0/1, true o false, Si o No)
- Aquí **no haremos registro de usuarios**, por lo que pueden insertar directamente estos valores en la base de datos con un par de usuarios para trabajar según los roles.
- El **Rol** posible de un usuario puede ser:
  - Administrador
  - Usuario Normal
  - Soporte Técnico
  - Analista

Esta será una tabla donde se almacenan estos datos con su identificador y nombre del rol, es una tabla referencial, emparejada mediante los identificadores.

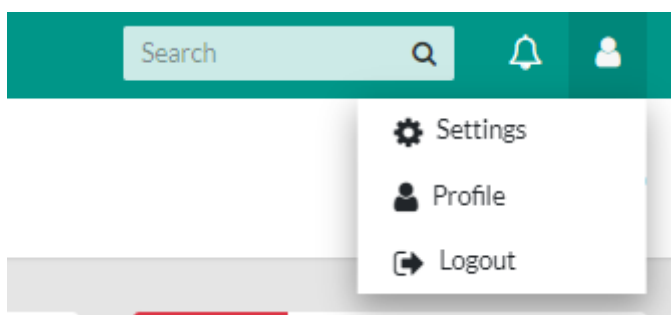
### Ingreso al Panel (index)

Tras el login correcto, nos ubicamos en el **index**. La pantalla principal se ve de esta manera, y los links del menú lateral izquierdo deben apuntar a los scripts respectivos.



Se debe **validar al inicio del index**, y **de cada página del panel**, que el usuario se encuentre logueado para poder visualizar el contenido, si no debe procesar lo necesario y redirigir al login.

Notar que **se muestran los datos del usuario y su nivel, junto a una imagen**. Estos datos deben estar en la sesión que se inicializó tras el login.



En la sección derecha del panel, pulsando en el icono de “usuario”, se despliega ese menú: se debe agregar en el Logout lo correspondiente a este proceso.

### Carga de Solicitud

Desde el menú lateral izquierdo, haciendo click en “Registro”, vemos este formulario, que toma los datos de la solicitud que el usuario desea cargar, con los campos de la template “registro.html”.

Los datos mínimos que se solicitan para cargar una solicitud son:

- ✓ Identificador único
- ✓ Título (cadena de hasta 100 caracteres)
- ✓ Descripción (cadena de hasta 1000)
- ✓ Usuario que la carga (usuario logueado)
- ✓ Fecha de carga (fecha actual)
- ✓ El tipo de solicitud: tomar datos de otra tabla referencial, con los valores, notar que está ordenado alfabéticamente:
  - Desarrollo de nuevas funcionalidades
  - Reporte de errores
  - Soporte técnico
- ✓ Fecha estimada de resolución: esto dependerá del tipo de solicitud:
  - Si la solicitud es del tipo *Desarrollo de nuevas funcionalidades*, la fecha estimada será de 1 semana posterior a la fecha de carga.
  - Para el tipo *Soporte técnico*, la fecha estimada será de 1 día posterior a la fecha de carga.
  - Para Reporte de errores. la fecha estimada será de 3 días posteriores a la fecha de carga.

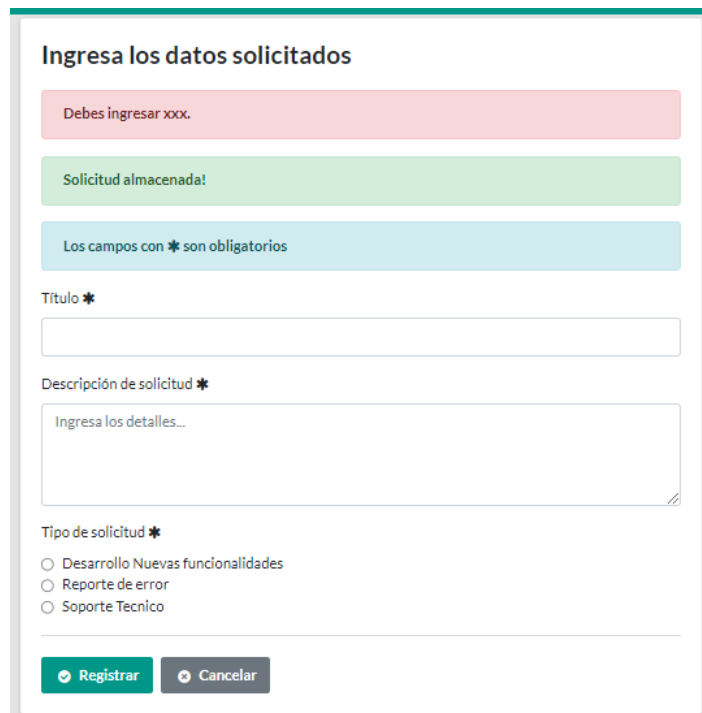
Para calcular esta fecha estimada, te ingeniarás la manera más práctica de poder hacerlo. Deberás usar una función propia que la calcule, considerando en su proceso lo anteriormente mencionado.

Dejo un link que puede servir, y un script test.php para que prueben.

<https://elinawebs.com/como-sumar-y-restar-fechas-con-php-con-strtotime-y-date/>

Aquí se validarán que **TODOS** los campos estén cargados para registrar la solicitud:

- Título y descripción deben contener como mínimo 5 caracteres cada uno.
- Elegir un tipo de solicitud (esta sección debe llenarse desde los datos de la tabla de manera ordenada alfab.)
- Informar con el mensaje correspondiente en caso que no se validen los datos.
  - Notar que la sección de Información (en celeste) debe estar siempre presente. Si hay errores mostrar la sección en rojo, si está todo ok mostrar el mensaje en verde. (Uno u otro)
  - El botón de cancelar lleva al Inicio (index)
- El resto de datos se deben almacenar internamente desde este proceso de registro.
- Aquí el botón también debe funcionar como se mencionó en el tip al principio.



Ingresa los datos solicitados

Debes ingresar xxx.

Solicitud almacenada!

Los campos con \* son obligatorios

Título \*

Descripción de solicitud \*

Ingresa los detalles...

Tipo de solicitud \*

☐ Desarrollo Nuevas funcionalidades

☐ Reporte de error

☐ Soporte Técnico

## Listado de Solicitudes

Haciendo click en el menú lateral izquierdo, en “Listados” >> “Listado de solicitudes”, llegamos a lo que vemos en la template “listado.html”.

- ✓ Aquí **dependiendo del rol del usuario** logueado, el listado podrá mostrar títulos diferenciados:

```
<div>
  <h1><i class="fa fa-th-list"></i> Listados</h1>
  <!-- si es administrador vera este titulo-->
  <p>Listado total de solicitudes</p>

  <!-- si es usuario normal vera este titulo-
  <p>Listado de mis solicitudes cargadas</p> -->

  <!-- si es analista o tecnico vera este titulo
  <p>Listado de solicitudes cargadas</p> -->
</div>
```

- ✓ Si es usuario Administrador: verá TODAS las solicitudes cargadas, puede ver todo, con el título "Listado total de Solicitudes"
- ✓ Si es Usuario Normal, verá solo las suyas cargadas (propias del usuario logueado), con el título "Listado de mis solicitudes cargadas".
- ✓ Si es rol Técnico o Analista, comparten el título "Listado de solicitudes cargadas".
  - El Técnico verá las solicitudes de tipo Soporte Técnico,
  - El Analista verá las otras (las de tipo Reportes de Errores y Desarrollo de nuevas funcionalidades)
- ✓ Entre paréntesis, mostrar en el titulo general la cantidad de registros que se visualizan en cada listado.
- ✓ Los registros deben verse ordenado por fecha/hora de creación en ascendente.

- ✓ Notar que **según sea el tipo de solicitud**, se visualiza el color de fondo de la fila:

- Las que sean de Soporte Técnico: viene dado por la clase de estilo "table-danger".
- Las del tipo de Reporte de Error: viene dado por la clase de estilo "table-warning".
- Las del tipo Desarrollo de nueva funcionalidad: viene dado por la clase de estilo "table-info".

```

135 <div class="col-md-12">
136 <div class="tile">
137 <h3 class="tile-title">Solicitudes (Nro Total)</h3>
138 <div class="table-responsive">
139 <table class="table">
140 <thead>
141 <tr>...
151 </tr>
152 </thead>
153 <tbody>
154 <tr class="table-danger">...
166 </tr>
167 <tr class="table-info">...
168 </tr>
180 <tr class="table-danger">...
182 </tr>
194 <tr class="table-warning">...
208 </tr>
209 <tr class="table-warning">...
210 </tr>
222 </tbody>
223 </table>
224 </div>
225 </div>
226 </div>
227 </div>
228

```

tipo

- ✓ La fecha del registro y la estimada, debe visualizarse en formato español con la hora tal como se ve. Puedes usar una función propia de PHP o una función tuya que lo genere.
- ✓ La descripción debe mostrar las primeras 20 palabras del contenido ingresado en este campo.

Listados  
Listado total de solicitudes

/ Listado / Listado De Solicitud

### Solicitudes (Nro Total)

#	Título	Descripción	Tipo	Registro	Fecha estimada	Solicitante	Opciones
1	El monitor enciende, pero con brillo muy bajo	Al encender el equipo, el monitor enciende pero con muy poco brillo (las primeras 20 palabras)...	Soporte tecnico	01/11/2023 10:23:56	02/11/2023 10:23:56	Carmen Ramirez	Ver detalles... Eliminar...
2	Modificar lista de precios	Generar cálculo automático de precio segun variación del dólar.	Desarrollo nueva funcionalidad	01/11/2023 12:55:56	03/11/2023 12:55:56	Gisela Marquez	Ver detalles... Eliminar...
3	Sin acceso a la VPN	Solicito me habiliten el ingreso a la red de la empresa	Soporte tecnico	01/11/2023 13:55:21	02/11/2023 13:55:21	Martin Cardozo	Ver detalles... Eliminar...
4	No muestra mensaje al fallar la clave de acceso	En acceso al sistema debe mostrar mensaje si hay fallo en la clave.	Reporte de error	02/11/2023 08:25:42	05/11/2023 08:25:42	Paola Torres	Ver detalles... Eliminar...
5	En el login, se ve mensaje	Al ingresar al equipo de trabajo con mi login y clave, veo error de conexion con la base de datos.	Reporte de Error	02/11/2023 14:30:15	09/11/2023 14:30:15	Mario Gimenez	Ver detalles... Eliminar...



### Consideraciones en el Desarrollo

- ✓ **Seccionar los scripts** en Encabezado, Menu lateral, Datos del usuario, Pie, Menu Derecho (el desplegable), funciones, usando una función de PHP que no permita continuar cargando el resto de código si ese script en cuestión no se encuentra.
- ✓ Deberás usar **Funciones Propias** para **conectar** a la base de datos, para **Validar** previo a la carga, para el **Registro**, y para el **Listado**. Dentro de cada una usarás las funciones necesarias de PHP para lograr lo requerido.
- ✓ Puedes usar la \$\_SESSION para lo que necesites
- ✓ Recuerda validar en cada script que el usuario se encuentre logueado mientras navega dentro del panel. Si no es así, se debe redirigir al login.
- ✓ Debes armar la base de datos relacional para poder almacenar y cumplir con lo requerido.
- ✓ Debes programar lo solicitado en este documento de requerimiento.
- ✓ Al terminar debes subir todo el comprimido nuevamente junto a la exportación de tu base de datos generada.
- ✓ Podras trabajar individualmente o de a dos.
- ✓ Recuerda mantener el diseño pautado.

### Detalles generales

- ✓ Trabaja tú mismo, leyendo, comprendiendo, pensando la lógica para codificar a tu manera. Analiza bien el enunciado, comprende lo que se pide. Luego pasa al desarrollo.
- ✓ Recuerda que debes comprender tu código para que en el coloquio puedas aclarar cómo lo hiciste.
- ✓ Puedes usar los ejemplos y ejercicios que vimos en las clases para poder resolver los requerimientos planteados, adaptando lo necesario, y consultando lo que tengas dudas. Puedes googlear y adaptar lo que necesites. Siempre y cuando sepas explicar motivos de tales usos.

### Puntuación

<b><u>Seccionado</u></b>	5%	
	Seccionar según lo requerido, la carga y el listado.	5%
<b><u>Login</u></b>	15%	
	Validación de datos ingresados según condiciones indicadas.	5%
	Actualización de fecha de ingreso	5%
	Generación de sesión de usuario y localizar en index	5%
<b><u>Panel gral.</u></b>	10%	
	Validación de sesión activa del usuario logueado.	4%
	Datos propios del usuario sección izquierda	3%
	Cierre de sesión	3%
<b><u>Registro</u></b>	30%	
	Validación de datos usando función en PHP.	10%
	Mensajes respectivos	5%
	Carga en la base con los datos solicitados	15%

<b>Listado</b>	40%	
	Títulos solicitados según rol del usuario logueado	5%
	Listado de datos según rol del usuario logueado	15%
	Coloreo de filas según tipo de solicitud.	5%
	Mostrar fecha y hora de carga y estimada, en formato español	10%
	Mostrar las primeras 20 palabras de la descripción.	5%

Porcentaje de aciertos	Calificación que le corresponde	Condición
0 a 29	1 (uno)	Desaprobado
30 a 49	2 (dos)	Desaprobado
50 a 59	3 (tres)	Desaprobado
60 a 64	4 (cuatro)	Aprobado
65 a 69	5 (cinco)	Aprobado
70 a 74	6 (seis)	Aprobado
75 a 79	7 (siete)	Aprobado
80 a 89	8 (ocho)	Aprobado
90 a 95	9 (nueve)	Aprobado
96 a 100	10 (diez)	Aprobado