

Integrantes del proyecto

Oscar Jorge Rapp (ojrapp@gmail.com)

Sergio Gabriel Giacomini (sgiacomini@gmail.com)

Guido Alfonsi (guidocompuserice@gmail.com)

Pablo Carrai (pablohernanjuan@hotmail.com)

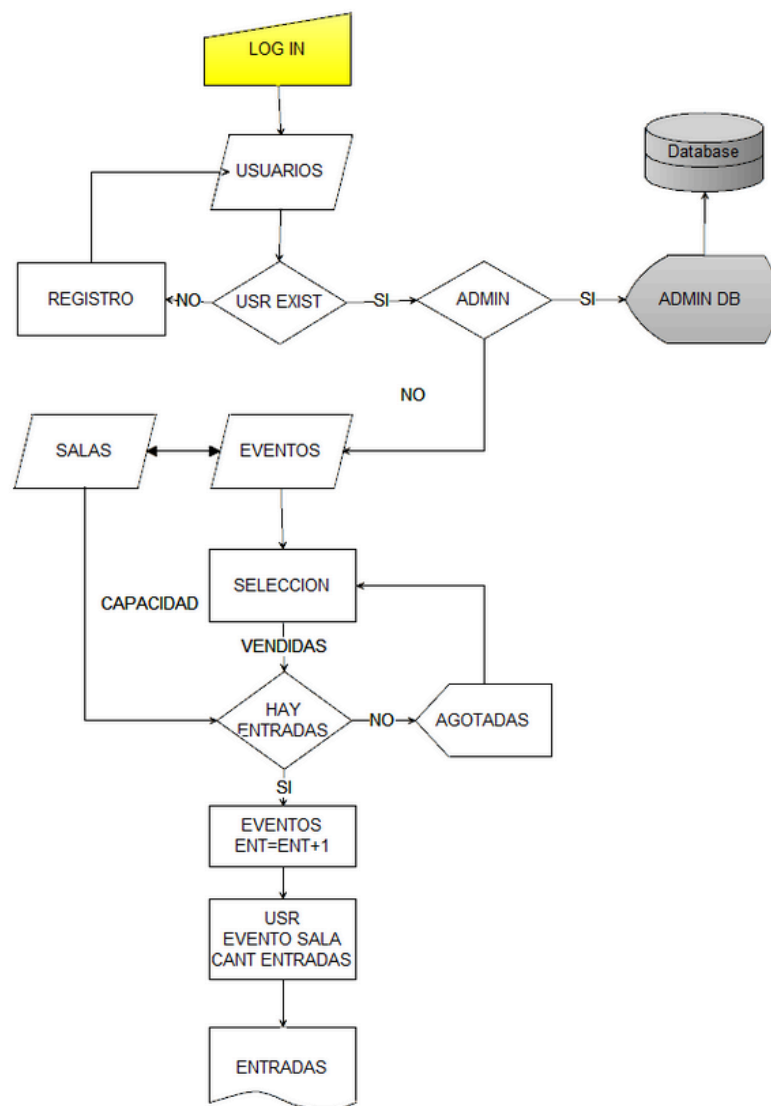
Objetivo del proyecto

Desde la cursada del Curso Introducción a la Programación en Python desarrollado por la Universidad de Vicente Lopez, con tutoría de It College. Se dio el puntapié que como trabajo integrador del curso en cuestión se genere un sistema de ticket de eventos. El puntapié inicial era tratar de aplicar muchos de los conceptos aprendidos en dicho curso y avanzar en cubrir una experiencia real en el desarrollo de un sistema web.

Github del proyecto.

<https://github.com/PabloCarrai/Ticketeadora-P-CVL>

Flujo aplicativo



Tablas de datos - Relaciones

Columnas

Nombre	Tipo	Tamaño
ID_BUT	Entero largo	
ID_ESP	Entero largo	
ID_SAL	Entero largo	
FILA	Entero	
BUTACA	Entero	
OCUPADA	Sí/No	
FECHA	Fecha con hora	
HORA	Fecha con hora	
USR	Texto corto	
F_CAMBIO	Fecha con hora	
H_CAMBIO	Fecha con hora	

Relaciones

ESP_ESPECTACULOSBUT_BUTACAS

ESP_ESPECTACULOS		BUT_BUTACAS
ID_ESP	1 ∞	ID_ESP

Attributes: Forzado; Actualizaciones en cascada; Eliminaciones en cascada
RelationshipType: Uno a varios

SAL_SALASBUT_BUTACAS

SAL_SALAS		BUT_BUTACAS
ID_SALA	1 ∞	ID_SAL

Attributes: Forzado; Actualizaciones en cascada; Eliminaciones en cascada
RelationshipType: Uno a varios

Columnas

Nombre	Tipo	Tamaño
ID_ESP	Entero largo	
ID_SALA	Entero largo	
TITULO_DEL_ESPECTACULO	Texto corto	
FECHA	Fecha con hora	
HORA	Fecha con hora	
ENTRADA_NRO	Entero	
PRECIO_UNITARIO	Moneda	
USR	Texto corto	
F_CAMBIO	Fecha con hora	
H_CAMBIO	Fecha con hora	

Relaciones

ESP_ESPECTACULOSBUT_BUTACAS

ESP_ESPECTACULOS		BUT_BUTACAS
ID_ESP	1	∞

Attributes: Forzado; Actualizaciones en cascada; Eliminaciones en cascada
RelationshipType: Uno a varios

SAL_SALASESP_ESPECTACULOS

SAL_SALAS		ESP_ESPECTACULOS
ID_SALA	1	∞

Attributes: Forzado; Actualizaciones en cascada; Eliminaciones en cascada
RelationshipType: Uno a varios

Columnas

Nombre	Tipo	Tamaño
ID_SALA	Entero largo	
NOMBRE_DE_LA_SALA	Texto corto	
CALLE	Texto corto	
NRO	Entero	
LOCALIDAD	Texto corto	
CAPACIDAD	Entero	
USR	Texto corto	
F_CAMBIO	Fecha con hora	
H_CAMBIO	Fecha con hora	

Relaciones

SAL_SALASBUT_BUTACAS

SAL_SALAS		BUT_BUTACAS	
ID_SALA	1	∞	ID_SAL

Attributes:

Forzado; Actualizaciones en cascada; Eliminaciones en cascada

RelationshipType:

Uno a varios

SAL_SALASESP_ESPECTACULOS

SAL_SALAS		ESP_ESPECTACULOS	
ID_SALA	1	∞	ID_SALA

Attributes:

Forzado; Actualizaciones en cascada; Eliminaciones en cascada

RelationshipType:

Uno a varios

TABLAS

ESP_ESPECTACULOS

ID_ESP INT AUTO_INCREMENT PRIMARY KEY,
ID_SALA INT,
TITULO_DEL_ESPECTACULO VARCHAR(255),
FECHA VARCHAR(10),
HORA VARCHAR(5),
ENTRADA_NRO INT,
PRECIO_UNITARIO DECIMAL(10, 2) NOT NULL

SAL_SALAS

ID_SALA INT AUTO_INCREMENT PRIMARY KEY,
NOMBRE_DE_LA_SALA VARCHAR(255),
CALLE VARCHAR(255),
NRO INT,
LOCALIDAD VARCHAR(255),
CAPACIDAD INT

USR_USUARIOS

id INT AUTO_INCREMENT PRIMARY KEY,
username VARCHAR(60),
password VARCHAR(60),
email VARCHAR(60)
tipo VARCHAR(10) (admin o cliente)

MODULOS

ABMCL DE TABLAS (usuarios tipo Admin)

Se accede desde la pantalla inicial al Log-in

Se valida como admin

Muestra admin.html

Usuarios

Bajas, modificaciones, y consultas.
html, flask, sql, phyton

Salas

Bajas, modificaciones, y consultas.
html, flask, sql, phyton

Espectáculos

Altas, bajas, modificaciones, y consultas.
html, flask, sql, phyton

EMISION DE TICKETS (Usuarios tipo clientes)

Selección de espectáculo.

Botón admin en esquina que envía al formulario log-in.

Log-in de usuario (cliente)

Si el usuario es válido

Ingresar cantidad de entradas.

Si el usuario no es válido

registro de usuario

Antes de emitir los tickets, validar capacidad de sala vs ENTRADA_NRO.

Debe cumplir: $ENTRADA_NRO + \text{cantidad de entradas} \leq \text{Capacidad de sala}$

Si hay disponibilidad sumar cantidad de entradas a ENTRADA_NRO

Emitir ticket

Usuario, espectáculo, fecha y hora, datos de la sala, entrada N°

Si la capacidad de sala fue cubierta,

mostrar mensaje "entradas agotadas"

1er etapa.

En esta etapa la idea era inicialmente tener listo un ABM de usuarios que permitiera registrar usuarios normales, y uno que permitiera crear administradores, que serían los encargados de dar de alta los eventos y a otros usuarios del mismo perfil.

Momentáneamente no se llegó a manejar perfiles en este ABM de usuarios. Si generamos un tipo especial de usuario que es el que puede manejar esas opciones. También tenemos disponible un ABM de eventos completamente funcional.

Flujo de trabajo.

Inicialmente para esta etapa el inicio de la aplicación está centrado en una pantalla de logueo que se ve como la imagen a continuación.



La imagen muestra una interfaz de usuario para el login de usuarios registrados. El fondo es un color azul claro. En el centro, hay un recuadro de color azul oscuro con el texto "Ingreso Usuarios Registrados" en blanco, con "Ingreso" y "Registrados" en una tipografía más grande y cursiva. Debajo de este recuadro, hay tres campos de entrada de texto con botones de envío integrados. El primer campo está etiquetado "Introduzca su Usuario", el segundo "Introduzca su Contraseña", y el tercero "Ingreso". En la parte inferior, hay un enlace que dice "Todavía no tiene una cuenta? [Regístrese aquí](#)".

En dicha página podemos loguearnos con un usuario ya existente, o en su defecto registrar nuestro propio usuario. En el caso de querer registrar nuestro usuario presionamos sobre la frase “Regístrate aquí” que nos mostrará la siguiente pantalla.

A screenshot of a registration form titled "Registración" in a dark blue box. Below the title are three input fields with placeholder text: "Introduzca su Usuario", "Introduzca su Contraseña", and "Introduzca su email". At the bottom are two buttons: "Registro" and "Página anterior".

Registración

Introduzca su Usuario

Introduzca su Contraseña

Introduzca su email

Registro

Página anterior

Completando las cajas de entrada de datos para el usuario, la contraseña y un correo electrónico, y presionando sobre el botón Registro generamos nuestro usuario. A continuación vemos un ejemplo de dicha pantalla con estos datos para un nuevo usuario.

A screenshot of the registration form with example data entered. The first input field contains "manuela", the second contains six dots representing a password, and the third contains "manuela@gmail.com". The "Registro" and "Página anterior" buttons are still present at the bottom.

Registración

manuela

••••••

manuela@gmail.com

Registro

Página anterior

Un detalle importante, es si el usuario existe tendremos un error. como se ve a continuación

Registración

La cuenta ya existe !

Introduzca su Usuario

Introduzca su Contraseña

Introduzca su email

Registro

Página anterior

En el caso de que el usuario sea nuevo veremos por pantalla el mensaje de que se ha creado el usuario, y nos dirigiremos a la pantalla de compra de entrada

Eventos

ID	Sala	Título	Fecha	Hora	Capacidad	Precio \$	Vendidas	Comprar
5	Teatro de Repertorio	El Padrino	26/08/2024	17:00	170	1570.00	33	Comprar
8	Cine York	La novicia No quiso hacerse la loca	12/08/2024	16:00	186	10000.00	41	Comprar
14	Teatro de Repertorio	Cascanueces	29/07/2024	21:00	85	15000.00	25	Comprar
17	mmm	Ohhhhh	12/08/2024	21:00	37	6969.00	13	Comprar
18	Teatro de Repertorio	Tiubron ataca	05/08/2024	21:00	44	7500.00	6	Comprar

Página anterior

Cerrar Sesión

En donde podremos comprar la entrada al show que queremos presenciar. Al presionar sobre comprar veremos la siguiente pantalla. En donde podremos elegir la cantidad de entradas y presionar sobre el botón comprar.

***Espectaculo
seleccionado***

Nombre de la Sala: Teatro de Repertorio

Espectaculo: El Padrino

Fecha: 26/08/2024

Hora: 17:00

Precio : \$ 1570.00

1

Comprar

Página anterior

Una vez hecho esto veremos la siguiente pantalla.

Compra

Gracias por su compra!

*Pronto recibirá en su casilla de correo el
detalle de la compra realizada.*

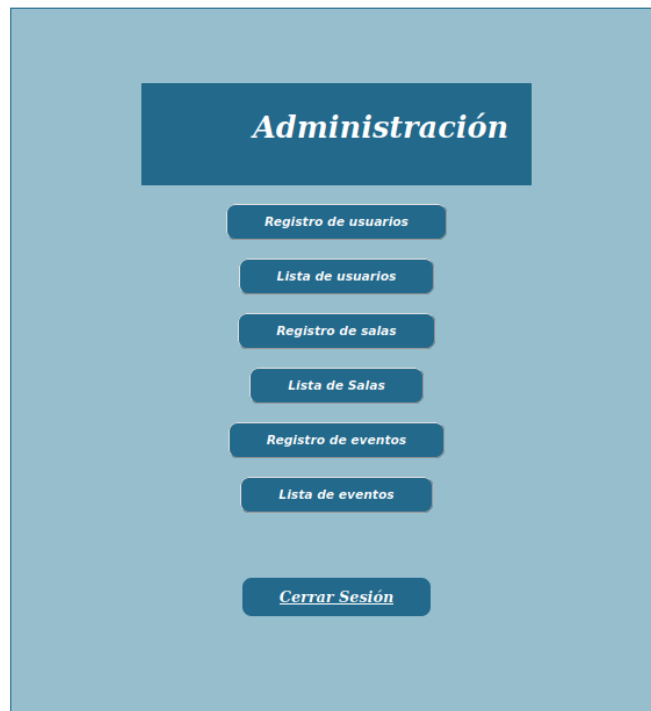
Volver

Cerrar Sesión

Presionando sobre cerrar sesión saldríamos del sistema. Y si presionamos sobre volver podremos seguir eligiendo la compra a otros espectáculos.

Etapas usuario administrador.

Al ingresar con una cuenta del tipo administrador* veremos un menu de opciones distintas que se ven del siguiente modo.



Registro de usuarios.

Este botón despliega la siguiente vista.



Como se puede apreciar es una pantalla de registro de usuario normal, con la diferencia en el último cuadro, en donde se puede dar de alta un usuario del tipo administrador si se ingresa la palabra admin, caso contrario(dejarlo en blanco) nos permite dar de alta un usuario normal.

Dependiendo de cómo procedamos con el alta de dicho usuario nos toparemos con la confirmación de la creación del mismo una vez presionado sobre el botón Registro.

Registración

Se ha registrado exitosamente !

Introduzca su Usuario

Introduzca su Contraseña

Introduzca su email

Tipo de Usuario

Registro

Página anterior

Listado de usuario

Al presionar sobre este botón tendremos una vista como se aprecia a continuación

Usuarios				
ID	Usuario	Password	mail	tipo
1	oscar	2024	ojrapp@ojrapp.com	admin
2	GuidoA	123456	guidoa@gmail.com	admin
5	GuidoB	123456	guidob@gmail.com	
6	GuidoC	123456	guidoc@gmail.com	
7	GuidoD	123456	guidod@gmail.com	
8	GuidoE	123456	guidoe@gmail.com	
9	Master	123456	master@gmail.com	admin
10	pablo	123456	pablo@nienpedo.com	admin
11	jorge	2024	jorge@gmail.com	
13	ruben	123456	ruben@ruben.com	
14	heman	123456	heman@heman.com	
15	GuidoZ	123456	guidoz@gmail.com	
16	GuidoY	123456	guidoy@gmail.com	
17	manuela	123456	manuela@gmail.com	
18	miranda	123456	miranda@gmail.com	
19	sergio	123456	sgiacomini@gmail.com	
20	skeletor	123456	skeletor@rul.com	

Registro de Salas

Al presionar sobre el botón Registro de salas, veremos un formulario como el que se muestra a continuación, que nos pedirá que ingresemos los siguientes datos. Nombre de la sala, Direccion Numero localidad y capacidad

salas

Sala CarapaShow

Independencia

345

Carapachay

344|

Registro

Página anterior

Lista de Salas

En esta sección podremos obtener un detalle de todas las salas que se dieron de alta. teniendo tanto el nombre de la sala, su dirección y la capacidad de la misma.

Salas

ID	sala	Domicilio	Nro	Localidad	Capacidad
1	Cine York	Italia	2024	Olivos	300
2	Teatro	Libertador	2024	Olivos	500
3	teatro Núcleo	Maipú	3143	Olivos	30
4	Teatro de Repertorio	Carlos F Melo	1756	Florida	30
5	mmmm	500	500	oli	50
6	Casa dela Cultura	Ricardo Gutierrez	1060	Olivos	150
7	Mariano Alcosta	Santa Fe	3345	Vicente Lopez	23

Página anterior

Registro de eventos

Esta sección sirve para dar de alta los eventos. Inicialmente tenemos disponible un desplegable con el listado de salas, y luego podemos ingresar el nombre evento, la fecha del mismo su horario la cantidad de lugares disponibles y el monto de la entrada

Eventos

Casa dela Cultura

novatos te dejan garpando

12/2/23

23

233

20000

Registro

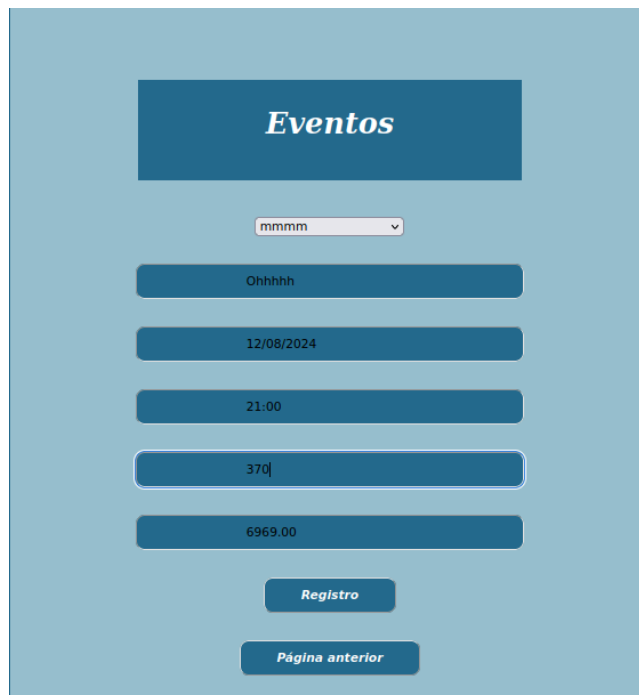
Página anterior

Lista de eventos

En esta sección podremos visualizar los eventos disponibles. Tambien se pueden editar y eliminar.

Eventos								
ID	sala	Título	Fecha	Hora	Capacidad	Precio \$	Vendidas	
5	Teatro de Repertorio	El Padrino	26/08/2024	17:00	167	1570.00	36	Eliminar Modificar
6	teatro Núcleo	IA GUERRA Y LA PAZ	20/07/2024	20:30	147	2000.00	10	Eliminar Modificar
8	Cine York	La novicia No quiso hacerse la loca	12/08/2024	16:00	1863	10000.00	41	Eliminar Modificar
13	teatro Núcleo	Pedro y el Lobo	12/07/2024	16:00	50	10000.00	4	Eliminar Modificar
14	Teatro de Repertorio	Cascanueces	29/07/2024	21:00	83	15000.00	27	Eliminar Modificar
16	Casa dela Cultura	Cuentos del Tio	20/07/2024	21:00	50	5.00	3	Eliminar Modificar
17	mmmm	Ohhhhh	12/08/2024	21:00	37	6969.00	13	Eliminar Modificar
18	Teatro de Repertorio	Tiubron ataca	05/08/2024	21:00	43	7500.00	7	Eliminar Modificar
19	Cine York	Los militares al desnudo	22/2/23	18	23	20000.00	0	Eliminar Modificar

Cuando modificamos algunos de los eventos veremos lo siguiente.



The screenshot shows a web form titled "Eventos" on a light blue background. The form contains several input fields and buttons:

- A dropdown menu with the text "mmmm" and a downward arrow.
- A text input field containing "Ohhhhh".
- A text input field containing "12/08/2024".
- A text input field containing "21:00".
- A text input field containing "370".
- A text input field containing "6969.00".
- A button labeled "Registro".
- A button labeled "Página anterior".

Armado de entorno para pruebas

En el caso de querer armar un entorno de pruebas para trabajar con el proyecto se recomienda utilizar entornos virtuales. En el proyecto existe un archivo que tiene lo necesario para armar el entorno virtual. Está dentro de la carpeta entorno_virtual

Ejemplo práctico de como armar el mismo. Inicialmente tenemos que tener instalado, un buen tutorial sobre como instalarlo en entornos Windows

<https://www.youtube.com/watch?v=N9PdRkXOQ4w>

En el caso de entornos linux

<https://www.youtube.com/watch?v=QyTXWZ0IJtI>

Teniendo esa base de un entorno virtual más el archivo de requirements.txt podrán hacer funcionar el proyecto. Tienen que tener en cuenta que el entorno virtual es una buena forma de tener las librerías necesarias instaladas para hacer funcionar la parte de python del proyecto. Otro requerimiento para este proyecto es tener un servidor con mysql. Para ello podrán instalar un servidor xamp en Windows

<https://www.youtube.com/watch?v=6cRoIPAXA2c>

Y en la carpeta de sql van a contar con un dump de la base actual con datos para probar la aplicación. Por otro lado la sección de Scripts_Auxiliares cuenta con varios archivos .py que tienen elementos para hacer consulta sobre las bases que se usan en el proyecto.

Pendientes

Compra de eventos

- no mostrar ID
- quitar botón página anterior
- página lista de eventos o posibilidad de poner filtros

Lista de usuarios

- quitar las contraseñas

Registro de salas

- validación al guardar

Listado de Salas

- Falta modificar y borrar

Registro de eventos

- validación al guardar (ingresar datos)
- validación al eliminar - Que no existan ventas para el evento
- validación al eliminar - pantalla de confirmación: botón eliminar con el mismo estilo que el resto.
- validación al modificar - si se modifica "Entrada número", en el caso de que tenga ventas, no podría ser menor a lo ya vendido.

Tabla eventos: Unificar campos "FECHA" y "HORA" en un solo campo de tipo DATETIME

Visualización de perfil propio.

- Modificación de datos personales o de perfil.
- Posibilidad de ver las entradas compradas, cancelar las mismas.
- recuperar contraseña

Los usuarios no se pueden modificar.

Salas no permite su modificación.

No hay un manejo de perfiles

solo hay usuario normal y admin

No existe un listado de entradas compradas. Tampoco hay un resumen de lo comprado ni el costo de lo consumido

Registro de Usuarios:

- Tipo de usuario, debiera ser un desplegable con los tipos predefinidos, sino se podrían crear infinidad de tipos, sin sentido
- Todos los campos dicen Introduzca...., menos Tipo de usuario

Lista de usuarios:

- Mail y Tipo, las iniciales están en minúsculas

Registro de Salas:

- Título en minúscula
- Mejorar Introduzca nombre de sala

Lista de Salas:

- sala en minúscula

Registro de Eventos:

-Quitar _

Lista de eventos:

-sala en minúscula

Modificar Eventos:

-El desplegable de Salas con estilo acorde al resto

-No hay tooltip

Eventos compra:

-Quitar botón Página anterior

-Más que entradas vendidas, no debiera informar entradas disponibles?

Espectáculo seleccionado:

-Reubicar selector de cantidad de entradas

Estructura del proyecto.

El proyecto cuenta con la siguiente estructura de archivos.

```
.
├── app.py
├── credenciales_ticketeadora.py
├── Documentacion
│   ├── Manual-Ticketeadora.pdf
│   └── README.me
├── entorno_virtual
│   └── requirements.txt
├── example-credenciales_ticketeadora.py
├── __pycache__
│   └── credenciales_ticketeadora.cpython-311.pyc
├── README.me
├── Scripts_Auxiliares
│   ├── borrado-de-tablas.py
│   ├── borra_registro.py
│   ├── borra_tablas.py
│   ├── crea la tabla espectaculos.py
│   ├── crea la tabla salas.py
│   ├── crea la tabla usuarios.py
│   ├── crea_db_ticketeadora.py
│   ├── crea_la_la_tabla_espectaculos.py
│   ├── crea-la-tabla-sala.py
│   ├── crea-la-tabla-usuario.py
│   ├── crea_tabla_ventas.py
│   ├── listado de datos de tablas.py
│   ├── listado tablas ticketeadora.py
│   ├── listado-de-datos-de-tablas.py
│   ├── listado-tablas-ticketeadora.py
│   └── lista_espectaculos.py
```



```
|
| |
| | — lista_salas.py
| | — lista_usuarios.py
| | — README.me
| | — registro_espectaculos.py
| — sql
|   |
|   | — createDbAndTable.sql
|   | — dump.sql
| — static
|   |
|   | — style.css
| — templates
|   |
|   | — consulta_elimina_espectaculo.html
|   | — consulta_usuarios.html
|   | — espectaculo_compra_confirmada.html
|   | — espectaculo_compra.html
|   | — home.html
|   | — index.html
|   | — index_ini.html
|   | — lista_espectaculos_compra.html
|   | — lista_espectaculos.html
|   | — lista_salas.html
|   | — lista_usuarios.html
|   | — login.html
|   | — register_comprador.html
|   | — register.html
|   | — registro_espectaculos.html
|   | — registro_salas.html
```

8 directories, 47 files

Clonación del repositorio.

Para tener disponible para pruebas o modificación del proyecto pueden obtener el mismo desde su repositorio oficial ubicado en

<https://github.com/PabloCarrai/Ticketeadora-P-CVL>

Haciendo un git clone sobre el mismo

git clone <https://github.com/PabloCarrai/Ticketeadora-P-CVL>