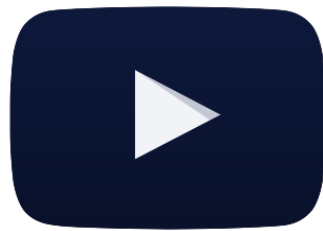




Escuela de Ingeniería en Computación

IC 6821 - Diseño de Software

Documentación del Prototipo



StreamTEC

Elaborado por:

**Yelson Andrés Monge Roa
2016089657**

12/06/2020

Introducción

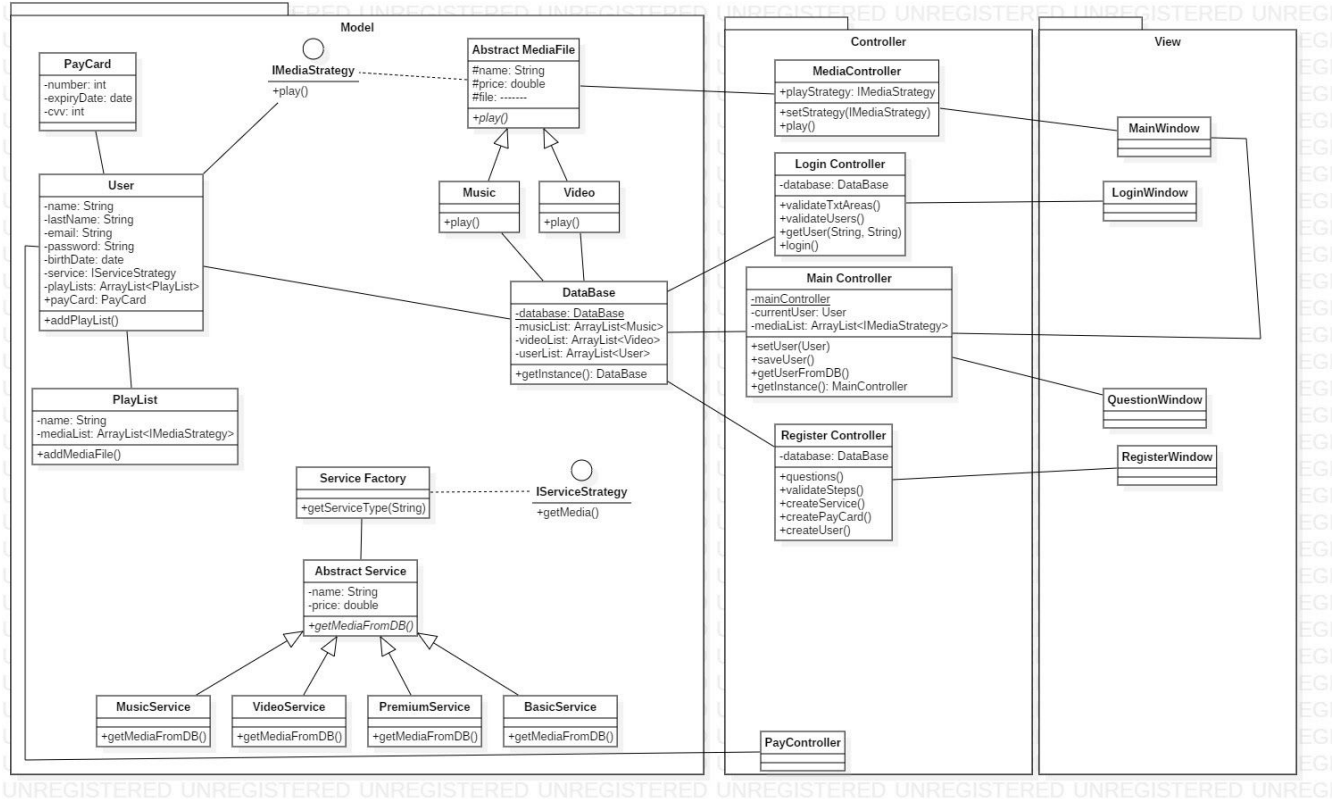
El siguiente documento tiene como propósito mostrar la solución propuesta hacia el problema del StreamTEC. A lo largo del documento veremos la explicación mediante diagramas, ejemplos y explicación redactada para que quede de la manera más clara posible, cuál fue la solución planteada. También veremos características que ayudan a mejorar la experiencia del usuario, tanto como UI, UX y accesibilidad. Y finalmente, explicaremos cómo ejecutar el programa y las librerías que se necesitaron utilizar, así como su debida instalación.

Problema

El departamento de Streaming del TEC StreamTEC, desea poner sus servicios a la venta, los servicios que van a vender son 4, uno de solo música y otro de solo video, el servicio VIP es el que contempla ambos, y un 4to servicio que es bajo demanda. Un usuario debe poder matricularse antes de poder comprar, para ello se requiere toda la información de contacto. Un usuario puede ver el catálogo de productos de acuerdo con su perfil. El usuario puede guardar los productos como en una lista de reproducción. En el mes del cumpleaños se le da un 5% de descuento, y en el día de su cumpleaños un usuario tiene acceso ilimitado.

Solución Planteada

El siguiente diagrama representa la solución planteada, cualquier eventualidad que sea complicado de ver, se adjuntará la imagen por separado en la carpeta del proyecto en donde estará este documento: documentation.



Es importante mencionar que la solución propuesta toma en cuenta que un usuario pueda tener varias playlist. Sin embargo, para mayor simplicidad, en el prototipo sólo se tiene una playlist por usuario.

Patrones Utilizados

La solución que se propone es crear un MVC que aisle por completo los datos, de la capa lógica y de la interfaz. Así le permite al sistema almacenar todos los datos necesarios sin necesidad de interactuar directamente con los datos, si no con sus respectivos controladores. Además el MVC le permite a los desarrolladores poder trabajar en la interfaz sin problema alguno.

Aparte del MVC, se plantearon e implementaron ciertos patrones de diseño que se consideró que podrían mejorar la calidad del software. Entre ellos:

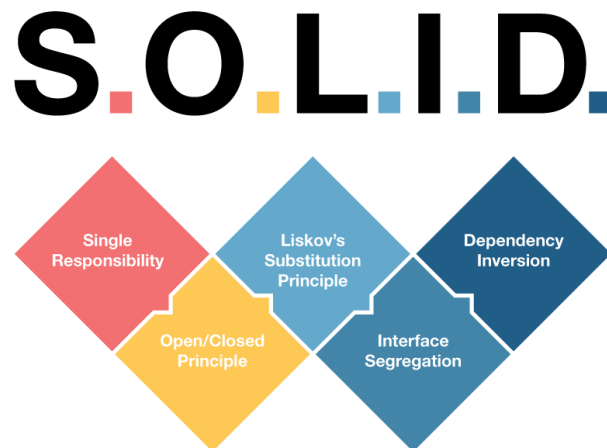
- **Strategy:** Este patrón se planteó para el momento de reproducir cada uno de los archivos, pensando en que en algún futuro podrían haber más tipos de archivos multimedia, no sólo música mp3 o archivos de video. Entonces este patrón acá me permite tener tranquilidad de que la aplicación puede reproducir cualquier archivo, sin importar cual sea la extensión. Además de que me permite no modificar el código del cliente directamente.
- **Factory+Strategy:** Estos dos patrones de diseño los implementé juntos, porque me permite trabajar con más facilidad tipo de servicios. Así podemos trabajar mejor a la hora de crear perfiles con su respectivo servicio, a la hora de ver qué contenido puede ver cada usuario, y así traer los archivos a la UI. También me permite no preocuparme por cómo se debe hacer el método de pago, pues en la estrategia se planteó el comportamiento del cobro. Así el cliente sólo debe llamar al usuario y decirle al servicio que cobre, ya el servicio por su cuenta realiza la estrategia de cobro.
- **Singleton:** Este lo implementé más que todo para la simulación de la base de datos, para que no haya más de una instancia y los datos siempre se mantengan consistentes. Y como criterio personal, siempre suelo hacer singleton con los controladores, ya que en ellos manejo las instancias de las ventanas y me da más seguridad el tener una instancia de cada controlador.

Principios SOLID

Los principios SOLID como tal, es algo que cuesta controlar con totalidad puesto que no estamos conscientes del todo cada vez que programamos, y sería complejo desarrollar un software que cumpla al 100% con estos principios. Sin embargo, utilizar patrones de diseño, nos ayuda mucho a cumplir con estos principios y sin darnos cuenta nos ayuda a mejorar la calidad del software. Vamos a explicar cómo nos ayudan los patrones que se implementaron a cumplir con SOLID.

El MVC nos permite modularizar la solución y cumplir con el Single Responsibility, así la capa de datos únicamente se encarga de almacenar, la capa de vista de mostrar estos datos, y en controlador de interactuar entre ellos. Además de que se programan controladores para cada tipo de sub problemas que hay que resolver.

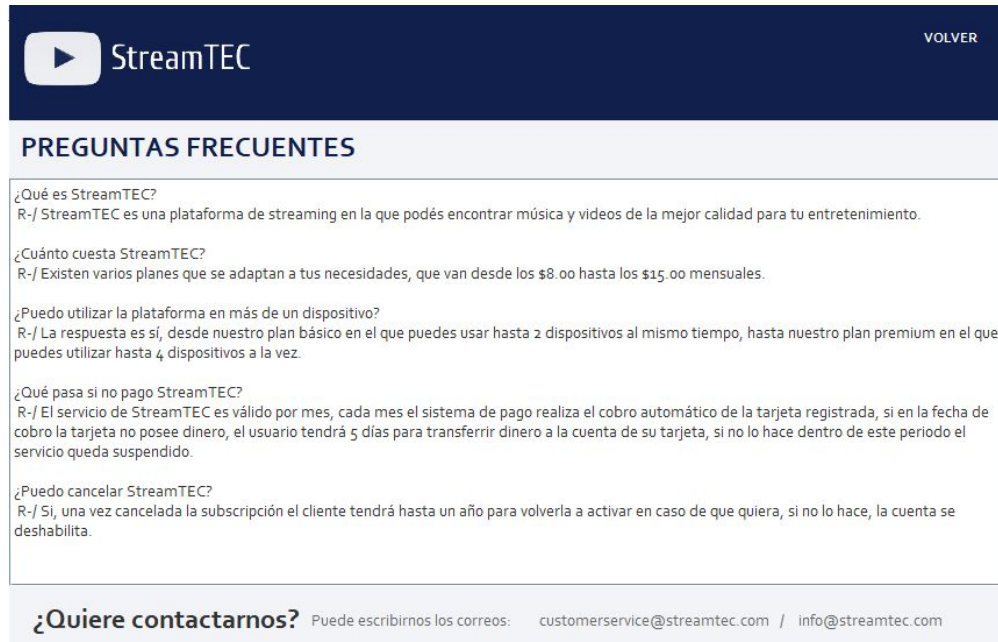
Los patrones de diseño Strategy y Factory nos permiten cumplir con el Open/Closed puesto que nos ayuda muchísimo a poder extender el sistema y adoptar nuevos cambios, sin necesidad de modificar mucho el código, para agregar una nueva funcionalidad basta con agregar una nueva clase que implemente las interfaces relacionadas al Strategy o al Factory (IMediaStrategy y IServiceStrategy en este caso).



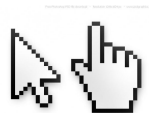
Accesibilidad

En esta sección, traté de implementar ciertas cosas que creo que pueden ser útiles para los usuarios finales de la aplicación.

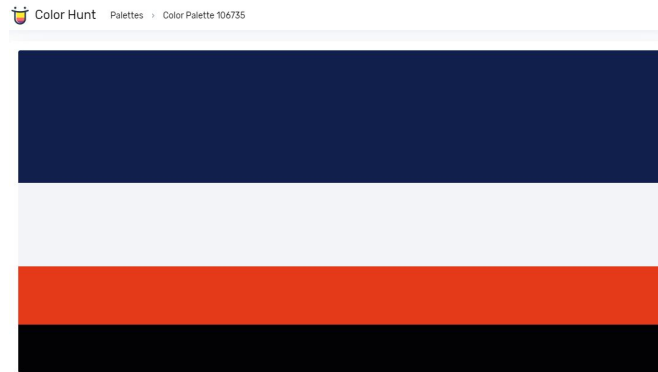
- **Preguntas Frecuentes:** Cree una sección en la cual los usuarios pueden ver las preguntas que suelen hacer lo demás usuario, y responder sus dudas sin perder mucho tiempo. El objetivo de esta sección es que sea más fácil para el cliente responder sus dudas.



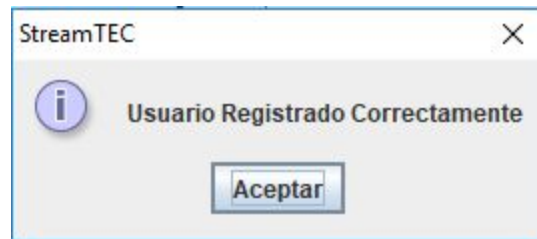
- **Cursor:** A veces es difícil en ciertas páginas o aplicaciones saber qué es un botón y qué no. Para evitar este tipo de confusiones, decidí configurar el cursor para que cambie del normal al de manita cuando pase sobre un botón.



- **Contrastes:** Elegí utilizar una paleta de colores que tuviera buen contraste y similar a los colores de Tecnológico, por lo que no será difícil identificar secciones, botones o texto entre un color claro con un color oscuro. Y quiero destacar que paleta tenía color rojizo naranja pero por gustos personales, decidí no utilizarlo.



- **Feedback:** Finalmente, hay un feedback constante pero no excesivo, se muestra el feedback únicamente, en cada paso en el que el usuario realiza una acción que modifique algún estado del sistema.



- **Ayuda Extra:** En los botones que son un ícono, se pusieron imágenes de acuerdo a lo que se va a realizar, botones intuitivos. Sin embargo, en el caso de que no se entienda del todo, cada botón de estos, tiene un texto de ayuda si se posa el cursor sobre el botón unos instantes.



Calidad de la Interfaz de Usuario

- **Separar el Registro:** Se optó por separar el registro de usuarios, para que sea más sencillo el poder llenar la información correctamente, ya que son muchos datos los que se le solicita al usuario.

REGISTRO DE USUARIO

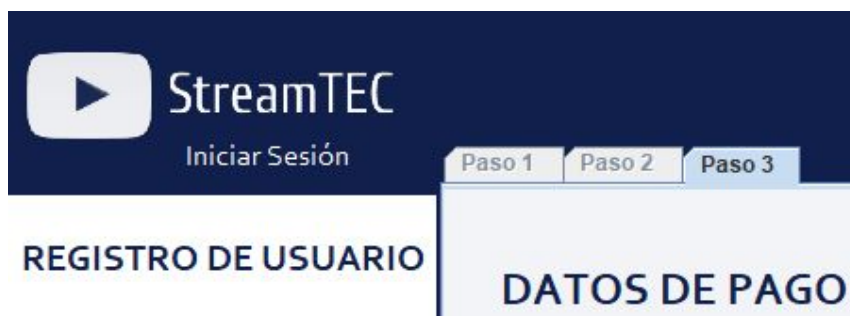
Pronto vivirás una experiencia única.

Sólo te quedan 3 sencillos pasos:

- ✓ Llena tus datos personales
- ✓ Elige tu plan favorito
- ✓ Confirma el método de pago

¡Listo!

- **Progreso Visible:** Se quería utilizar un progress bar pero por motivos de tiempo y simplicidad no se implementó, sin embargo, durante el registro, el usuario puede ver en qué paso se encuentra.



- **Color Adecuado de Botones:** Finalmente, tenemos color de los botones, decidí utilizar un contraste en los botones para mostrarle al usuario cuál acción es la principal y cuál es la que invalida, cancela o se desvía del flujo normal.





Otras Consideraciones

La intención principal del programa era tener un list para escoger la canción o video a reproducir, pero por motivos de tiempo se hizo en un combobox. El programa está desarrollado en Java, en el IDE 8.0.2 de NetBeans, por lo que se

Manual de Usuario

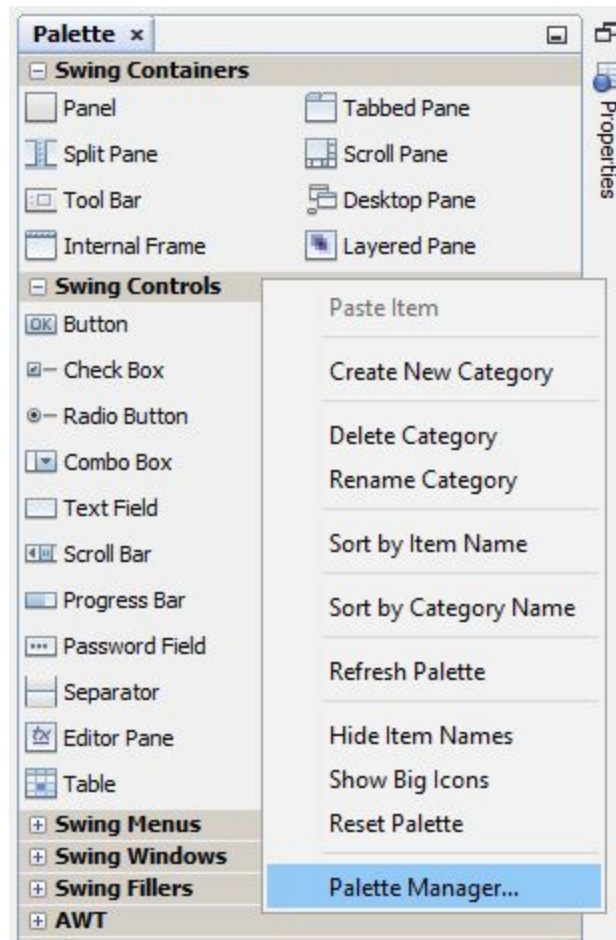
El programa está desarrollado en Java, en el IDE 8.0.2 de NetBeans, por lo que se recomienda ejecutar allí. Pero con cualquier otro entorno que soporte Java debería servir. Para ejecutarlo basta con tener el proyecto descargado, abierto en el IDE y presionar el botón de "Run".

Yo utilicé una librería en el proyecto, para solicitar formatos de fechas, y tener la posibilidad de elegir la fecha y no ingresar los datos manualmente. Yo ya incluí en el proyecto la librería, y se supone que ya debería de servir, pero por si acaso no funciona, acá está el manual de cómo instalarla.

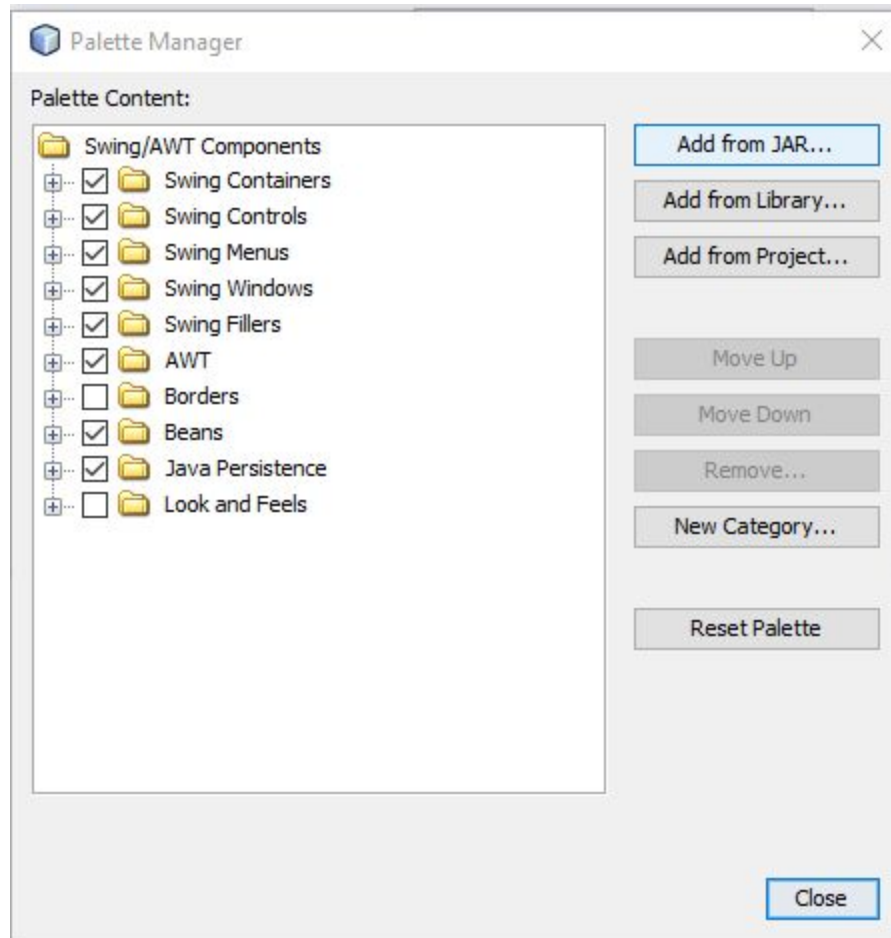
Hay que descargar la librería jcalendar desde el link:

<http://www.java2s.com/Code/Jar/j/Downloadjcalendar14jar.htm>

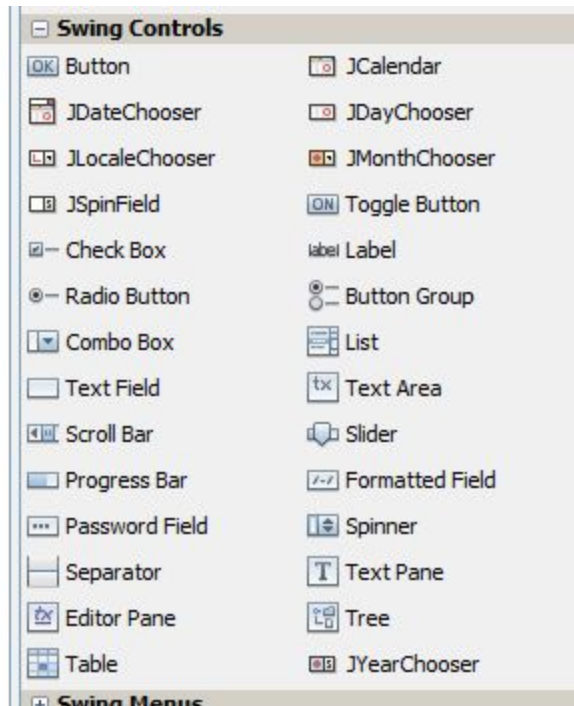
Cuando ya está descargada, descomprimos el zip y el jar file lo incluimos en el proyecto. Luego vamos a la parte del diseño de la interfaz de **windowRegister** y damos click derecho sobre la paleta de **Swing Controls** y seleccionamos la opción **Palette Manager**:



Luego, debemos darle a la opción **Add from JAR**



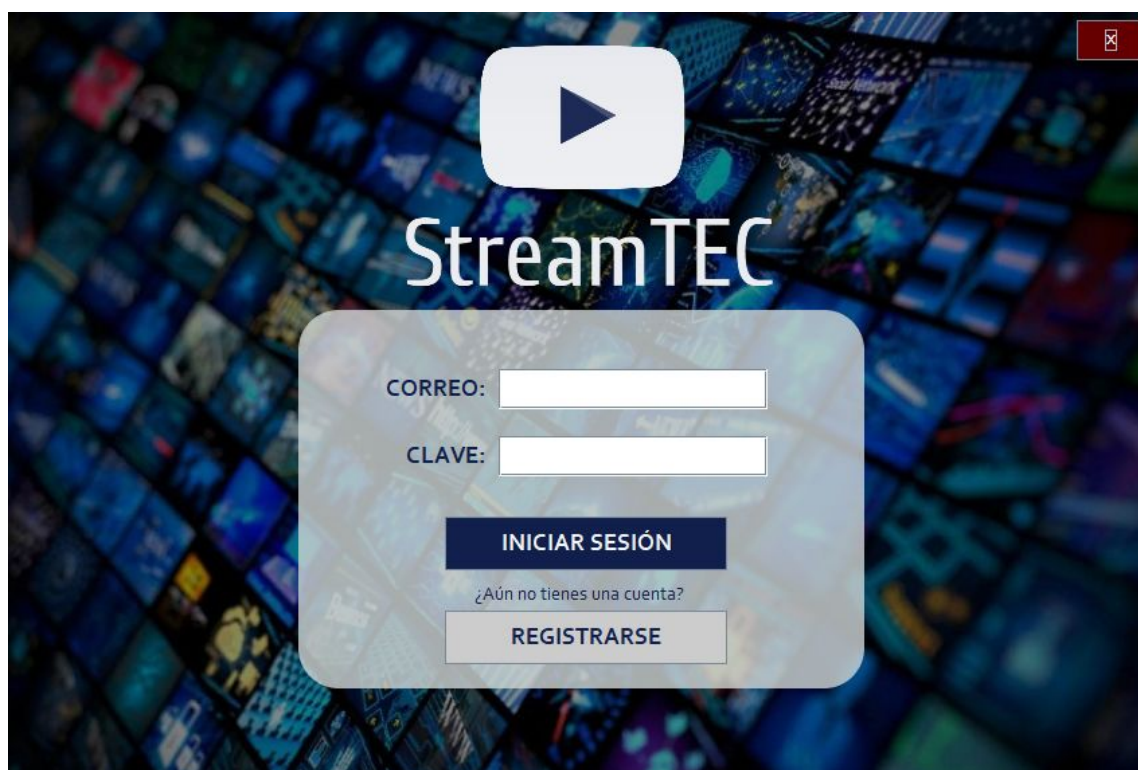
Luego buscamos el path del JAR, lo seleccionamos y le damos **Next**, y finalmente nos va a aparecer una ventana con los componentes que queremos agregar, seleccionamos todos y luego escogemos la carpeta **Swing Controls** para que nos aparezcan ahí.



Debe registrar un usuario para poder empezar a utilizar el programa, sin embargo, ya hay unos datos precargados para probar los inicios de sesión, las playList, los cobros y el acceso ilimitado. Para iniciar sesión con cada uno de los perfiles hice una tabla:

Correo	Contraseña	Tipo de Cuenta
yelsonmonge@gmail.com	1234	Premium
video@gmail.com	1234	Video
musica@gmail.com	1234	Musica
basic@gmail.com	1234	On Demand

A continuación una vista de las pantallas.



Premium

Lose Yourself - Eminem	▲
Mockingbird - Eminem	
Bohemian Rhapsody - Queen	
La Cosa Nostra - Blon	
Don't Stop Me Now - Queen	
Mentiras y Azar - Walls	
Chuty vs Nitro	
GOAT Messi	▼

Lose Yourself - Eminem ▼

Reproducir

Agregar a Mi Lista

Reproduciendo:

StreamTEC

Iniciar Sesión

Paso 1

Paso 2

Paso 3

REGISTRO DE USUARIO

Pronto vivirás una experiencia única.

Sólo te quedan 3 sencillos pasos:

✓

Llena tus datos personales

✓

Elige tu plan favorito

✓

Confirma el método de pago

¡Listo!

Con StreamTEC disfruta de la mejor calidad en streaming del mundo

Preguntas Frecuentes

DATOS PERSONALES

Nombre:

Apellido:


Correo Electrónico:

Contraseña:

Confirmar:

Fecha de Nacimiento:

CONTINUAR

StreamTEC

VOLVER

PREGUNTAS FRECUENTES

¿Qué es StreamTEC?

R-/ StreamTEC es una plataforma de streaming en la que podés encontrar música y videos de la mejor calidad para tu entretenimiento.

¿Cuánto cuesta StreamTEC?

R-/ Existen varios planes que se adaptan a tus necesidades, que van desde los \$8.00 hasta los \$15.00 mensuales.

¿Puedo utilizar la plataforma en más de un dispositivo?

R-/ La respuesta es sí, desde nuestro plan básico en el que puedes usar hasta 2 dispositivos al mismo tiempo, hasta nuestro plan premium en el que puedes utilizar hasta 4 dispositivos a la vez.

¿Qué pasa si no pago StreamTEC?

R-/ El servicio de StreamTEC es válido por mes, cada mes el sistema de pago realiza el cobro automático de la tarjeta registrada, si en la fecha de cobro la tarjeta no posee dinero, el usuario tendrá 5 días para transferir dinero a la cuenta de su tarjeta, si no lo hace dentro de este periodo el servicio queda suspendido.

¿Puedo cancelar StreamTEC?

R-/ Si, una vez cancelada la suscripción el cliente tendrá hasta un año para volverla a activar en caso de que quiera, si no lo hace, la cuenta se deshabilita.

¿Quiere contactarnos?

Puede escribirnos los correos: customerservice@streamtec.com / info@streamtec.com