

DESARROLLO DE INTERFACES

2º DAM

I.E.S. POLITÉCNICO H. LANZ
JOSÉ MARÍA MOLINA



TEMA 2-7 – LIBRERÍAS FX

2-7 LIBRERÍAS FX

Existen muchas librerías que se basan en JavaFX y amplían su funcionalidad, tanto para crear la interfaz, representar los datos o para crear otras funcionalidades extra.

Vemos algunas de ellas en el proyecto [\[javafx-ejemplos-libreríasFX\]](#) aunque este código no está en moodle porque es parte de la práctica (si se proporciona no se podría tener en cuenta para evaluar. A final de curso se pondrá público)

✓ 1 PORTALES DE RECURSOS

✓ 2 LIBRERÍAS FX

1 – PORTALES DE RECURSOS

✓ **JPRO:** <https://www.jpro.one/> **JPRO**

– una de las plataformas que existen para ejecutar JavaFX online (despliegue en web: SEGUNDO TRIMESTRE)

– Demos de librerías: <https://www.jpro.one/showcase>

✓ **JFX-CENTRAL :** <https://www.jfx-central.com/>



– Se ejecuta gracias a JPRO

– Centro de recursos: ejemplos, documentación, librerías, etc... <https://www.jfx-central.com/libraries> (ej curiosos: Countries, Medusa, PDFViewFX, etc)

2 – LIBRERÍAS FX

✓ Bibliotecas de Interfaz de Usuario

En azul los que veremos (o hemos visto) funcionando

- **ControlsFX**: Amplía el conjunto de controles estándar de JavaFX. (ya lo hemos visto en el tema 2-4 mediante los Validators).
- **CalendarFX**: proporciona un calendario gráfico muy vistoso y configurable.
- **TilesFX**: proporciona un conjunto de controles gráficos muy visuales en forma de tile (mosaico) orientados a la presentación de datos.
 - <https://javadoc.io/doc/eu.hansolo/tilesfx/21.0.3/index-all.html>
 - <https://github.com/HanSolo/tilesfxdemo> (Código para probar)
- **Medusa**: proporciona un conjunto de medidores gráficos.
<https://github.com/HanSolo/medusa> (Código para probar)

2 – LIBRERÍAS FX

✓ **Librerías de Material Design:** temas, estilos, controles, notificaciones, etc...

- **JFoenix:** Ofrece un aspecto Material Design (de google) a las aplicaciones JavaFX.
- **MaterialFX:** Alternativa a JFoenix.
- **CaspianFX:** Proporciona temas predefinidos y personalizables. Sigue los estilos básicos de JavaFX pero personalizables.
- **ModernLook:** Ofrece una amplia gama de temas y estilos.
- **DialogsFX:** Simplifica la creación de cuadros de diálogo personalizados.
- **SweetAlertFX:** Implementa notificaciones SweetAlert en JavaFX.
- Etc...

2 – *LIBRERÍAS FX*

- ✓ **Bibliotecas de Gráficos y Visualización**
 - **ChartsFX:** Crea gráficos y visualizaciones de datos.
 - **JavaFX Graphics:** Crea gráficos y visualizaciones de datos.
 - **FXyz:** Permite crear escenas 3D interactivas.
 - **SciChart:** Ideal para gráficos de alto rendimiento y grandes conjuntos de datos.
 - **JFreeChart:** Biblioteca popular para crear gráficos 2D.
 - **ProcessingFX:** Combina JavaFX con el lenguaje Processing para gráficos interactivos.

2 – LIBRERÍAS FX

✓ Otras Bibliotecas Útiles

- ~~GMapsFX~~: Librería que permite consultar la API de Google Maps (necesita APIKey vía registro en Google Cloud) OBSOLETA: en su lugar utilizar un WebView y configurar la API de GOOGLE.
- **JavaFXPorts**: Permite ejecutar aplicaciones JavaFX en dispositivos móviles.
- **TestFX**: Facilita la creación de pruebas automatizadas para interfaces de usuario JavaFX.
- **Gluon Mobile**: Framework para desarrollo móvil multiplataforma.
- **FXGL**: Biblioteca para desarrollo de juegos 2D en JavaFX. Permite Game Objects (entities), físicas (PhysicsComponents) y colisiones (collision handlers). <https://github.com/AlmasB/FXGLGames>

2 – LIBRERÍAS FX

✓ Librerías probadas (Nov.2025):

```
dependencies {  
    implementation 'org.controlsfx:controlsfx:11.2.2'// 1. controlsfx (Controles UI adicionales)  
    implementation 'eu.hansolo:tilefx:21.0.9'// 2. tilefx (Tiles para Dashboards)  
    implementation 'io.fair-acc:chartfx:11.3.1' // 3. chartfx (Librería de Gráficos de Alto Rendimiento - io.fair-acc)  
    implementation 'com.github.almasb:fxgl:21.1'// 4. fxgl (Motor de Juegos JavaFX) Incompatible con JavaFX+21  
    implementation 'com.calendarfx:view:12.0.1'// 5. calendarfx (Componente de Calendario/Agenda)  
}
```