

## Manual Técnico del Desarrollador - FacturaPro

### ## 1. Arquitectura y Estructura del Proyecto

El programa sigue una estructura básica que separa los datos, la lógica de almacenamiento y la interfaz de usuario.

- **Estructura de Carpetas (Maven):**
    - src/main/java: Contiene todo el código fuente de las clases .java.
    - src/main/resources: Contiene recursos como las imágenes (/img/logo.jpeg).
  - **Componentes Lógicos Principales:**
    - **Clases de Datos (Modelo):** Definen la estructura de la información.
      - Producto, Cliente, DetalleFactura, Factura, Empresa.
    - **Clases de Persistencia (Storage):** Se encargan de leer y escribir en los archivos.
      - ProductoStorage, ClienteStorage, UserStorage, ConfiguracionManager, GarantiaStorage, DevolucionStorage.
      - Toda la data se guarda en la carpeta del usuario: **%APPDATA%/FacturaApp**. La clase PathsHelper centraliza todas las rutas a estos archivos.
    - **Clases de Interfaz Gráfica (GUI):** Todas las clases que terminan en GUI y construyen las ventanas. FacturaGUI es la ventana principal que orquesta a las demás.
    - **Clases de Lógica y Seguridad:**
      - LicenseManager: Valida la licencia al inicio.
      - UserSession: Mantiene la sesión del usuario logueado.
      - PasswordManager: Hashea y verifica las contraseñas de los usuarios.
      - BackupManager: Contiene la lógica para crear y restaurar copias de seguridad.
-

## ## 2. Flujo de Trabajo de Licenciamiento (Tu Rol como Vendedor)

Este es el proceso completo para generar y distribuir licencias.

### ### 2.1. Preparación Única: Creación de tus Claves Maestras

Este paso **se hace una sola vez** para establecer tu "firma" digital.

1. **Ejecuta la Herramienta:** Compila y ejecuta la clase LicenseGenerator.java por separado. Es tu herramienta de desarrollador, no es parte del programa del cliente.
2. **Elige la Opción 1:** En la consola, selecciona la opción 1 para "Generar nuevo par de claves".
3. **Resultado:** Se crearán dos archivos:
  - private.key: **Tu llave privada y secreta.** Guárdala como si fuera oro y haz una copia de seguridad. **Nunca la compartas.**
  - public.key: Tu llave pública.
4. **Copia la Clave Pública:** La consola te mostrará una larga cadena de texto en formato Base64. Esta es tu clave pública. Cópiala por completo.
5. **Incrusta la Clave en el Programa:**
  - Abre el archivo LicenseManager.java en tu proyecto principal.
  - Busca la constante PUBLIC\_KEY\_STRING.
  - Pega la clave que copiaste entre las comillas, reemplazando el texto de ejemplo.
6. **Recompila tu Programa:** Vuelve a compilar tu aplicación (mvn clean package) para que la nueva versión contenga la clave pública incrustada.

Ahora tu aplicación está lista para verificar las licencias que solo tú puedes crear.

### ### 2.2. Proceso por Cliente: Generar una Licencia

Este proceso **se repite para cada cliente** al que le vendas el software.

1. **Obtén el ID de Máquina:** Pídele a tu cliente que ejecute el programa. Como no tendrá licencia, le aparecerá un error mostrando su "**ID de máquina**". El cliente debe enviarte este ID.
2. **Ejecuta la Herramienta de Nuevo:** Vuelve a ejecutar LicenseGenerator.java.

3. **Elige la Opción 2:** Selecciona la opción 2 para "Crear una licencia para un cliente".
  4. **Ingresa los Datos:** La herramienta te pedirá:
    - Nombre del cliente.
    - El **ID de máquina** que el cliente te proporcionó.
    - Los días de validez de la licencia (ej. 365).
  5. **Genera el Archivo:** Se creará un archivo license.key en la misma carpeta. Este archivo es único para ese cliente y esa máquina.
  6. **Entrega la Licencia:** Envíale el archivo license.key a tu cliente. Él deberá colocarlo en la carpeta %APPDATA%/FacturaApp. Al volver a abrir el programa, este se iniciará correctamente.
- 

### ## 3. Puntos Clave del Código y Lógica Interna

- **Sistema de Permisos:** La lógica de roles se centra en el método `UserSession.isAdmin()`. Este método simplemente comprueba si el nombre del usuario logueado es admin. La clase `FacturaGUI` usa este método en `aplicarPermisos()` para habilitar o deshabilitar los `JMenuItem`.
  - **Clave Administrativa:** Se gestiona en `ConfiguracionManager`. El método `getAdminPassword()` lee la clave del archivo `config.properties` (con un valor por defecto de "admin123"). Cualquier función crítica (eliminar cliente, editar stock, etc.) llama a este método para verificar la clave ingresada por el usuario en un `JOptionPane`.
  - **Centralización de Rutas:** Nunca escribas una ruta de archivo directamente en el código. Usa siempre la clase `PathsHelper` para obtener la ruta correcta a cualquier archivo de datos.
- 

### ## 4. Compilación y Distribución

1. **Compilar:** Desde la terminal, en la raíz del proyecto, ejecuta `mvn clean package`.

2. **Distribuir:** El archivo a entregar al cliente se encuentra en la carpeta target/ y se llama sistema-facturacion-1.0.0-jar-with-dependencies.jar (el nombre puede variar según tu pom.xml).
  3. **Paquete Final:** Lo que le entregas a tu cliente es:
    - El archivo .jar ejecutable.
    - El archivo license.key que generaste específicamente para él.
    - El manual de usuario final.
- 

## ## 5. Solución de Problemas Comunes del Desarrollador

- **Recuperar Acceso de Administrador:** Si el usuario admin se borra del archivo usuarios.csv manualmente, nadie podrá acceder a las funciones administrativas. La solución es **eliminar el archivo usuarios.csv** de la carpeta %APPDATA%/FacturaApp. La próxima vez que se inicie el programa, se recreará el usuario admin con la contraseña por defecto 1234.
- **Resetear Clave Administrativa:** Si olvidas la clave para editar stock/eliminar clientes, ve a %APPDATA%/FacturaApp, abre config.properties y borra la línea que empieza con admin.password=.... La clave volverá a ser admin123.
- **Perdí mi private.key:** Si pierdes tu llave privada, deberás repetir el **Paso 2.1** para generar un nuevo par. Sin embargo, esto invalidará todas las licencias que hayas emitido anteriormente. Deberás contactar a todos tus clientes, enviarles una nueva versión del programa (compilada con la nueva clave pública) y generarles una nueva licencia con tu nueva clave privada. **¡Haz una copia de seguridad de tu private.key!**