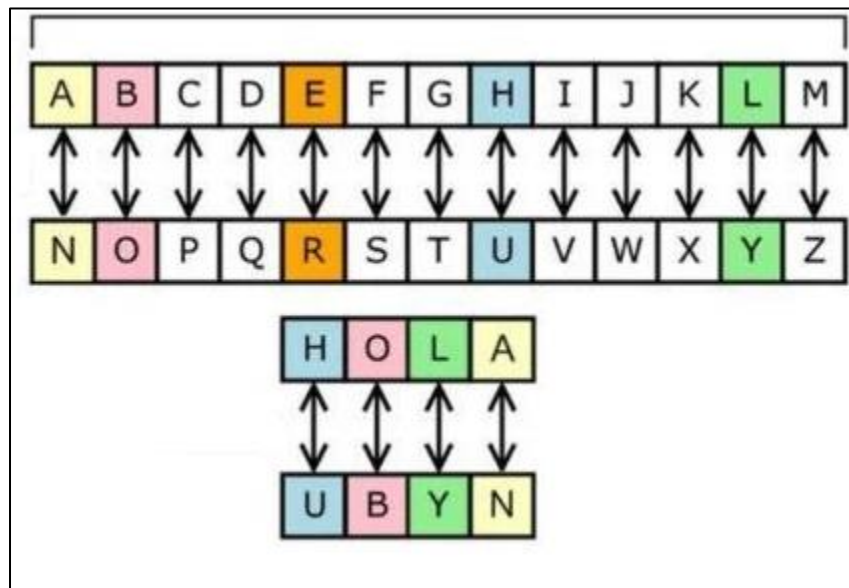


DOCUMENTACION DE PROYECTO – CIFRADO CESAR

1. Descripción General

Este proyecto implementa el cifrado César en Java, permitiendo al usuario cifrar o descifrar textos ingresados manualmente o leídos desde archivos .txt. Utiliza una estructura modular con separación de lógica (CifradoCesar), definición de alfabeto (Alfabeto como Enum) y una interfaz de consola (Main).



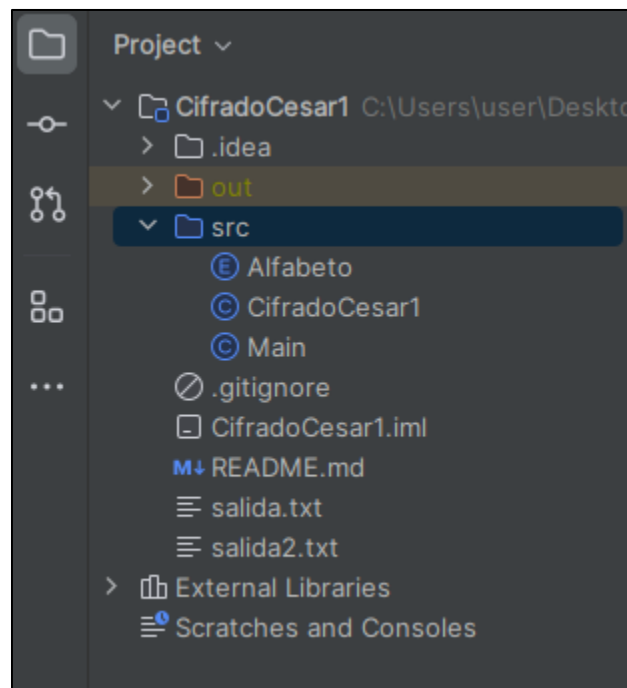
2. Estructura del Proyecto

CifradoCesar/

- └─ Alfabeto.java ← Enum con letras A-Z
- └─ CifradoCesar.java ← Lógica de cifrado y descifrado
- └─ Main.java ← Menú y control del flujo
- └─ salida.txt / salida2.txt ← Archivos de salida generados automáticamente

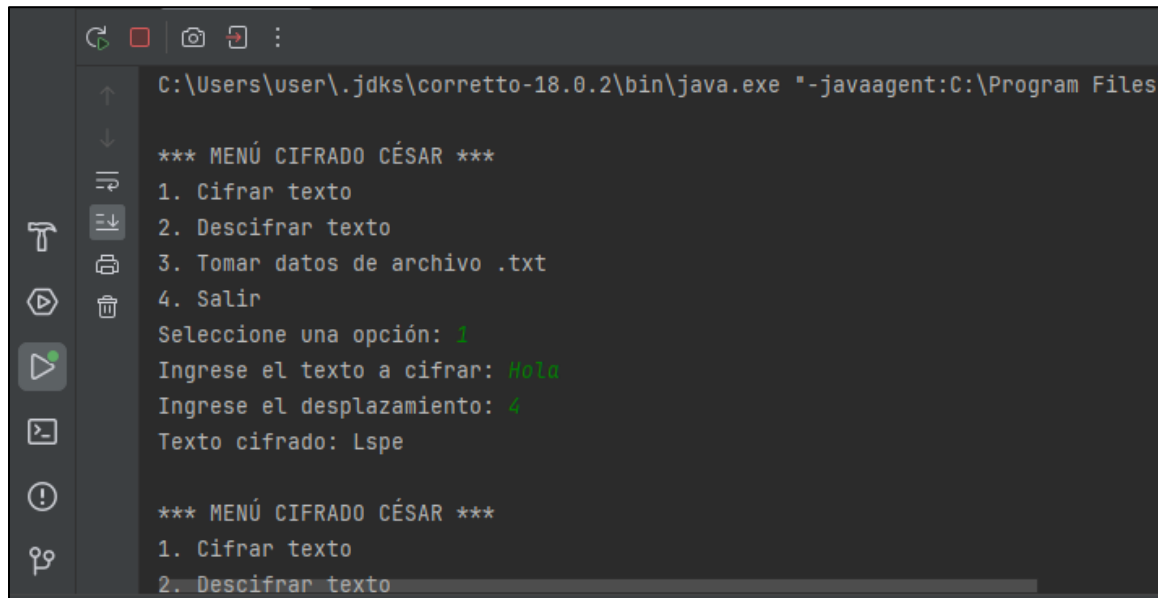
3. Clases y Componentes

- Alfabeto.java: Enum con los caracteres A-Z. No contiene métodos, solo define el alfabeto válido.
- CifradoCesar.java:
 - encrypt(String text, int shift): Cifra el texto respetando el alfabeto definido.
 - decrypt(String text, int shift): Descifra el texto aplicando desplazamiento inverso.
 - buscarIndice(...): Devuelve la posición de una letra en el enum.
- Main.java:
 - Menú interactivo por consola.
 - Opción 1: Cifrar texto desde teclado
 - Opción 2: Descifrar texto desde teclado
 - Opción 3: Leer archivo .txt, procesar y guardar como salida.txt, salida2.txt, etc.
 - Opción 4: Salir del programa
 - generarNombreArchivoSalida(): Crea nombres únicos para los archivos de salida.



4. Flujo del Programa

[Inicio] → Menú → (1: Cifrar | 2: Descifrar | 3: Archivo | 4: Salir) → Procesar → Mostrar/Guardar Resultado → Volver al Menú



```
C:\Users\user\.jdk\corretto-18.0.2\bin\java.exe "-javaagent:C:\Program Files'  
  
*** MENÚ CIFRADO CÉSAR ***  
1. Cifrar texto  
2. Descifrar texto  
3. Tomar datos de archivo .txt  
4. Salir  
Seleccione una opción: 1  
Ingrese el texto a cifrar: Hola  
Ingrese el desplazamiento: 4  
Texto cifrado: Lspe  
  
*** MENÚ CIFRADO CÉSAR ***  
1. Cifrar texto  
2. Descifrar texto
```

5. Algoritmo de Cifrado César

Para cada carácter del texto:

- Si está en el enum: convertir a mayúscula, encontrar su índice, aplicar shift, obtener nueva letra, devolverla en mismo caso.
- Si no está en el enum: mantener el carácter original.

6. Casos de Prueba

Texto: "ABC" → Shift: 3 → Cifrado: "DEF"

Texto: "XYZ" → Shift: 3 → Cifrado: "ABC"

Texto: "Hola Mundo" → Shift: 1 → Cifrado: "Ipmb Nvoep"

Texto: "Ipmb Nvoep" → Shift: 1 → Descifrado: "Hola Mundo"

Texto: "Texto con 123!" → Shift: 5 → Cifrado: "Yjcyt hts 123!"

Archivo: contiene "Abc xyz" → Shift: 2 → Resultado: "Cde zab" (guardado en salida.txt o similar).

7. Requisitos Técnicos

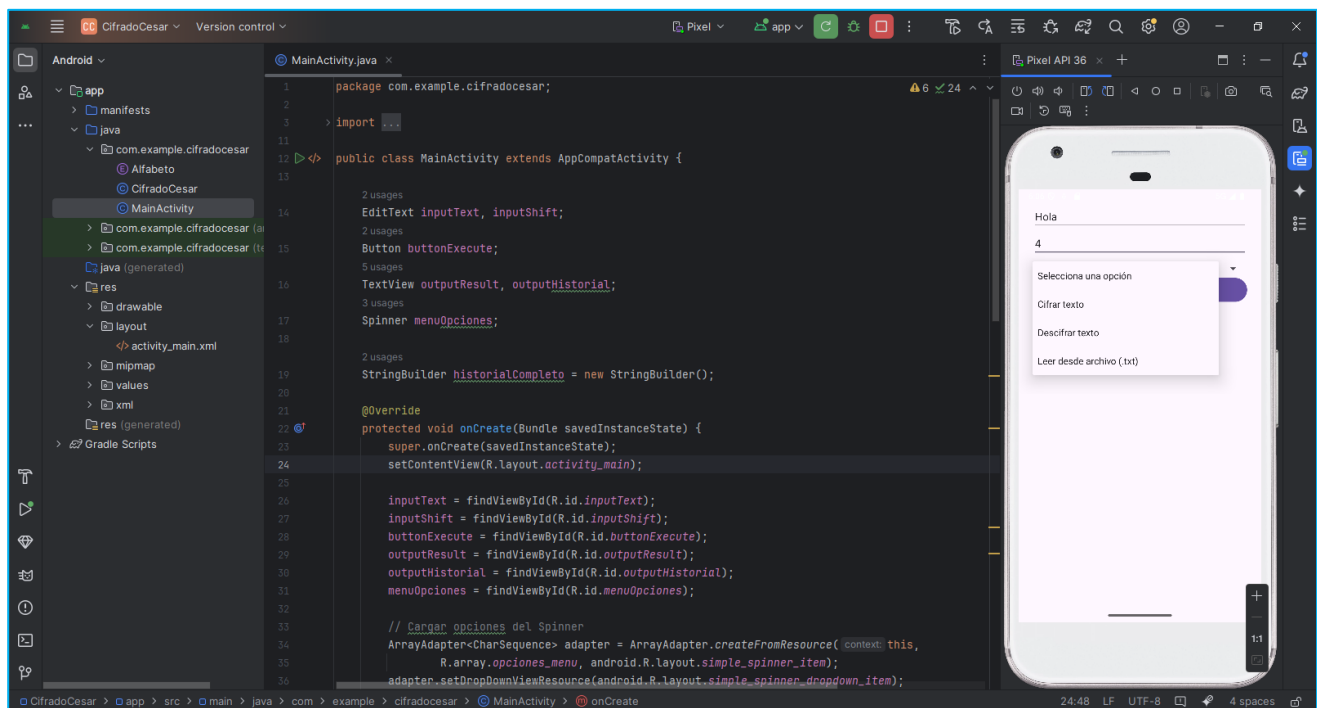
- Java SDK 8 o superior (actualmente probado con JDK 18)
- IDE como IntelliJ, Eclipse o VS Code
- Permisos de lectura/escritura en la carpeta del proyecto

8. Front End – Android Studio App

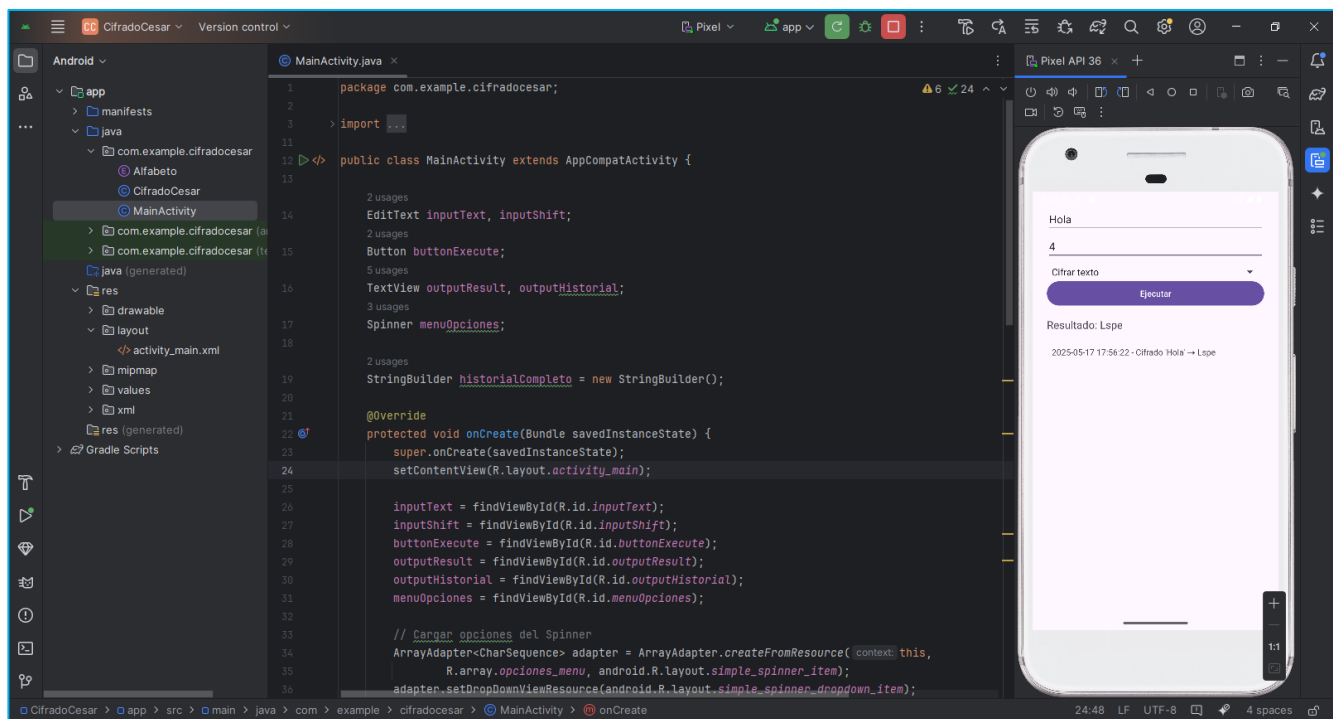
Esta aplicación móvil implementa una interfaz amigable que permite a los usuarios cifrar y descifrar texto usando el algoritmo clásico de Cifrado César. La app está diseñada para ser utilizada de forma intuitiva desde un teléfono Android, con campos sencillos y un historial visible de resultados.

La pantalla principal permite al usuario ingresar el texto, elegir el desplazamiento y seleccionar la operación a realizar (cifrado o descifrado) desde un menú desplegable. El resultado se muestra de inmediato y se guarda en un historial visual dentro de la misma sesión.

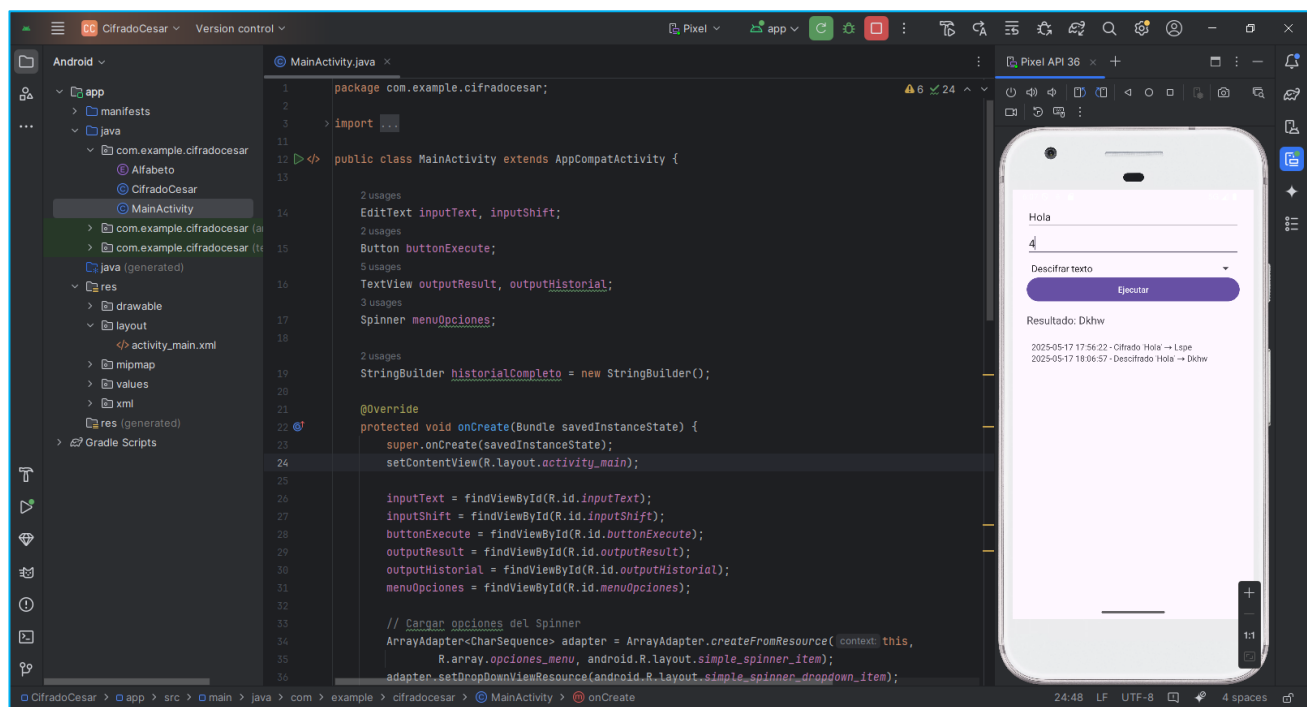
Actualmente, la app está completamente funcional para cifrado y descifrado manual. La opción de lectura desde archivo se encuentra marcada como 'en desarrollo', permitiendo planificar versiones futuras con manejo de archivos externos.

The image is a screenshot of the Android Studio IDE. On the left, the 'Project' view shows the file structure of the 'CifradoCesar' app, including 'manifests', 'java' (with 'com.example.cifradocesar'), 'res', and 'Gradle Scripts'. The 'MainActivity.java' file is open in the center editor. The code defines the MainActivity class, which extends AppCompatActivity. It includes fields for EditText (inputText, inputShift), Button (buttonExecute), TextView (outputResult, outputHistorial), and Spinner (menuOpciones). The onCreate method initializes these views and sets up a Spinner with a custom dropdown item. On the right, the 'Preview' window shows a mobile app interface. It has a text input field containing 'Hola', a shift value input field containing '4', and a dropdown menu with options: 'Selecciona una opción', 'Cifrar texto', 'Descifrar texto', and 'Leer desde archivo (.txt)'. The app is running on a virtual Pixel 3a device.

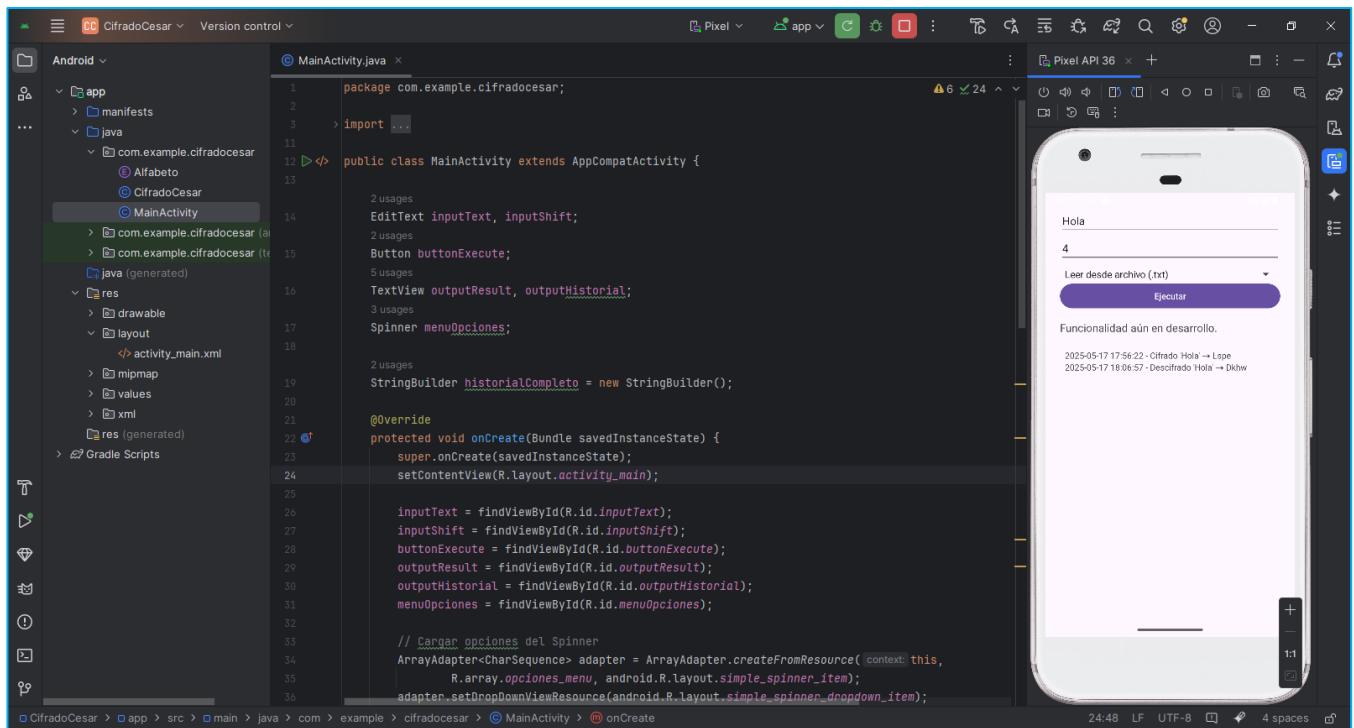
Muestra el Menú desplegable



Se muestra el cifrado para el texto ingresado



Se muestra el descifrado para el texto ingresado



Se muestra la opción de tomar datos aun en desarrollo; para la versión futura de manejo de archivos desde la APP.