Project Report: File Compressor & Decompressor

Introduction

File management often requires reducing storage space or transferring large sets of files efficiently. Compression is a widely used solution that reduces file sizes by eliminating redundancies, while decompression restores them to their original form. This project, **File Compressor & Decompressor**, provides a desktop application that allows users to **compress multiple files into a ZIP archive** and **decompress ZIP files back to their original form**. It is designed to be lightweight, user-friendly, and uses only **core Java libraries**, ensuring compatibility across platforms without additional dependencies.

Abstract

The **File Compressor & Decompressor** project is implemented in **Java** using the `java.util.zip` package. It allows users to select multiple files for compression into a ZIP archive or extract files from an existing ZIP. The application features a **graphical user interface (GUI)** built with Java Swing, making it accessible to non-technical users. It also includes a **progress bar** to track long operations and a **logging system** to maintain records of compression and decompression activities. The tool is designed for students, professionals, and anyone needing a simple solution for file compression tasks without installing third-party tools.

Tools Used

1. **Java (JDK 17)** – Programming language for implementation.
2. **Java Swing** – For creating the graphical user interface.
3. **`java.util.zip` package** – Provides `ZipOutputStream` and `ZipInputStream` for compression and decompression.
4. **VS Code IDE** – Used for writing and running the project.

Steps Involved in Building the Project

1. **Requirement Analysis**

○ Determine key features: file selection, ZIP compression, decompression, progress tracking, and logging.

2. **UI Design**

○ Build a simple desktop interface using **Swing**, with buttons for compression and decompression, and a **progress bar**.

3. **Compression Module**

○ Use `ZipOutputStream` to write multiple files into a single ZIP archive.

○ Implement a callback function to update progress during compression.

4. **Decompression Module**

○ Use `ZipInputStream` to read and extract files from an existing ZIP archive.

○ Handle directory creation and restore original file structures.

5. **Logging System**

○ Write operation details (e.g., number of files processed, output ZIP name, timestamp) to a log file `compression_log.txt`.

6. **Testing & Debugging**

○ Test with various file types and sizes.

○ Ensure error handling for invalid ZIPs, large files, and permission issues.

Conclusion

The **File Compressor & Decompressor** project successfully delivers a cross-platform desktop tool for file compression and decompression using only **core Java libraries**. With its **intuitive interface, progress tracking, and logging features**, it provides an efficient solution for everyday file management. This project demonstrates practical applications of **Java Swing** for UI development and the **`java.util.zip` package** for handling ZIP operations. It can be extended in the future to support **folder compression**, **password-protected ZIPs**, and **advanced compression formats**.