ATOZFILE – SMART FILE ORGANIZER

1. Introduction

In the modern digital world, users frequently deal with large numbers of unorganized files such as images, videos, documents, and software packages. Manual file organization is time-consuming, inefficient, and error-prone. There is a strong need for an automated system that can intelligently manage files and maintain a history of operations.

AtoZfile is a system-level desktop application developed in C using the GTK framework that automatically scans folders, categorizes files based on their types, and organizes them into structured directories. The application also integrates a database system to store history and provides a statistics dashboard to monitor file operations.

2. Problem Statement

Users often store thousands of mixed files in a single directory, leading to poor file management, difficulty in searching, risk of data loss, and reduced productivity. Existing solutions are either too complex, heavy, or lack transparency in file operations.

3. Objectives of the Project

The main objectives of this Smart File Organizer are:

- To design a smart file management system using low-level programming.
- To automatically organize files into category-based folders.
- To implement both **Move** and **Copy** operation modes.
- To store operation history using a database.
- To develop a clean and responsive GUI using GTK.
- To provide real-time logs and a statistics panel.
- To ensure safe file handling with backup and undo support.

4. Scope of the Project

The scope of AToZfile includes:

- Organizing files such as images, videos, audio, documents, archives, executables, and code files.
- Supporting dynamic folder creation
- Tracking every operation through a SQLite database.
- Displaying statistics and live activity logs.
- Running as a standalone desktop application on Windows (and portable to Linux).

Future enhancements may include cloud integration, AI-based file classification, duplicate file detection, and advanced search filters.

5. Existing System

In the existing system:

- File organization is mainly done **manually** using traditional file explorers.
- Users have to **create folders manually** and move files one by one.
- There is **no automatic categorization** of files based on type.
- High chance of **human errors** such as misplacing or deleting files.
- **No real-time progress or activity logs** during file transfer.
- Managing large numbers of files is **time-consuming and tedious.**
- Users cannot easily view **statistics** about their file organization.
- Many advanced tools are **heavy, complex, or not beginner-friendly.**

6. References

- GTK Documentation – *GTK Developer Guide* (official manual for building GUI applications using GTK library).
- SQLite Documentation – *SQLite official documentation for embedding database in C*. (https://sqlite.org)
- MSYS2 and GCC Compiler Documentation (official guides for building and deploying C applications).
- YouTube Tutorials (GTK & SQLite in C):
    - *GTK 3 GUI in C – beginner GTK application guide* — video on making a basic GTK GUI in C.
    - *GTK in C | GUI Implementation* — tutorial on building GUI using GTK in C.
    - *SQLite basics and integration with C* — video showing SQLite use in C projects.
- GitHub Examples & Code Resources:
    - *GTK+ 3 Tutorial in C* — practical GTK3 examples in C. (https://github.com/321zer0/gtk-c-tutorial)
    - *GTK Examples* — collection of GTK examples in C and other languages (https://github.com/Miqueas/GTK-Examples)
- AI framework help for program and project structuring and modularity.

7. Software and Hardware Requirements

Software Requirements

- **Operating System:** Windows 10 / Windows 11
- **Programming Language:** C
- **GUI Framework:** GTK 3
- **Database:** SQLite
- **Compiler/Build Tools:** GCC (via MSYS64)
- **Development Environment:** Any C-supported editor/IDE (VS Code, Code:Blocks, etc.)
- **(Extra nothing is required as it a deployed windows application and all other functionality and libraries are bundled with the installer , it's a install and run kind of application)**

Hardware Requirements

- **Processor:** Intel i3 or above
- **RAM:** 4 GB or more (8 GB preferred)
- **Hard Disk:** 400 MB minimum
- **System Type:** 64-bit computer
- **Display:** 1366 × 768 resolution or higher

8. Conclusion

AtoZfile is a smart desktop file organizer developed in C using GTK and SQLite that provides an efficient solution to manual file management problems. The application automatically categorizes files, supports move and copy operations, maintains operation history through a database, and displays real-time logs and statistics through a user-friendly graphical interface. The project successfully demonstrates the practical use of low-level programming, file system handling, GUI development, and database integration in building a real-world deployable desktop application, helping users save time, reduce errors, and keep their systems well organized.