**DOCUMENTATION TECHNICAL / DSI ProSoft**

**1. Introducing EasySave**

EasySave 2.0 is a Windows application with a graphical user interface (WPF) developed in C# on .NET 8.0. This version considerably improves the user experience, offering an intuitive interface while retaining the power and flexibility of the console version.

Allowing file backups to be performed on different types of media (local, external and network disks).

Thanks to its command-line interface for choosing the runtime, EasySave offers a fast, efficient solution for securing your files.

**What's new in version 2.0**

WPF graphical interface: more ergonomic and accessible.

Choice of log format (JSON/XML) before execution of unlimited backups.

Enhanced file encryption management with encryption time tracking.

Optimized execution if an application is already running, blocking the interface.

Optimized logs and real-time status.

Can be used via an .exe executable.

**Target users**

Professionals wish to secure their critical files.

Home users are looking for a simple, effective backup solution.

System administrators need a fast, automated tool.

**Prerequisites**

- System: Windows with .NET 8.0 installed.

- Access rights: Read/write on source and target directories.

- Disk space: Check available space before launching a backup.

**2. Installation and configuration**

1. Use the executable provided in the zip file “bin/Release/net8.0/Publisch.zip”.

2. Unzip the files in a directory of your choice.

3. Click on the EasySave.exe application to start using it.

The application works immediately after installation!

**3. Using EasySave**

Creating a backup job

A backup job is a task stored in the software, enabling files to be copied from one location to another.

Each backup job is defined by the following elements:

- Backup name: A unique identifier used to identify the backup.

- Source directory: Folder containing the files to be backed up.

- Target directory: Destination where the files will be copied.

- Backup type:

**Full: Copies all files on each run.**

**Differential: Copies only files modified since the last full backup.**

Running a backup

Select the backups to run.

Select log format (JSON/XML).

Business software check: Backup is blocked if critical software is running.

Start execution: Real-time monitoring via the interface.

File encryption after backup (if enabled).

Display of backup and encryption logs.

EasySave runs backup jobs created from the GUI via the command line, and supports several execution modes:

* To run a single job:

Enter the index of the backup to run, e.g. 1-3’ to automatically run backups 1 to 3:

1

à Launches backup **no.1 only.**

* To run several backups in sequence:

Enter the index of the backup to run, e.g. 1-3’ to automatically run backups 1 to 3:

1-3

à Runs backups **1, 2 and 3 sequentially.**

* To run several backups in parallel:

Enter the index of the backup to run, e.g. 1-3’ to automatically run backups 1 to 3:

1;3

à Simultaneously launches **backups 1 and 3.**

**4. Types of Backups**

* Full Backup

🔹 **Description**: Copies all files from the source directory to the target directory.  
🔹 **Advantage**: Easy to restore, ensures complete copy.  
🔹 **Disadvantage**: Longer and more disk space intensive.

* Differential Backup

🔹 **Description**: Copies only files modified since the last full backup.  
🔹 **Advantage**: Saves time and disk space.  
🔹**Disadvantage**: Requires a full backup to be restored correctly.

**5. Managing logs and reports**

Logs are managed according to the format selected just before execution!

Daily log file:

EasySave records all its actions in a JSON/XML log file updated in real time.

Information contained:

- Timestamp of each action.

- Name of the backup concerned.

- Source and destination address of files.

- Transfer size and duration (in milliseconds).

- Encryption time of selected copied files before execution.

- Any errors.

🔹 File format: JSON/XML  
🔹 Example file Name: backup\_log\_2024-02-10.json/xml

Real-time status file:

In addition to logs, EasySave generates a real-time status file containing the progress of backups.

Logged data:

- Backup name

- Last action timestamp

- Status (Active, Not Active)

- Progress: complete%, files remaining, size of remaining files

- Address of file being transferred

- Encryption time of selected copied files before execution.

🔹 File format: JSON /XML  
🔹 Example file Name: backup\_state.json/xml

**6. Best practices and recommendations:**

🔹 Avoid spaces in file paths to avoid errors.

🔹 Do not run backups at the same time as business software.

🔹 Daily log files are stored in: C:\ProgramData\CESI\EasySave\Logs

🔹 Real-time log files are stored in the project directory: bin/Relase/net8.0/Publish

🔹 The file in which the various backups created are saved is stored in the same directory as the project solution (.sln file). It is called “RegisteredBackups.txt”.

🔹 Check the available disk space on the target directory.

🔹 Check log files regularly to anticipate possible errors.

**7. Troubleshooting and FAQs**

Common problems and solutions

| **Problem** | **Solution** |
| --- | --- |
| **Encryption does not work** | Check that the file is one of the selected extensions. |
| **Files are not saved** | Check that the source and target directories exist and are accessible. |
| **The log file is empty or missing** | Make sure EasySave has written access rights. |