# COMPREHENSIVE TECHNICAL DOCUMENT AND END-USER GUIDE FOR DOCCENTRIK

**Introduction**

Welcome to the **DocCentrik Centralized Document Processing Tool**. This document serves as both a technical reference and a user guide to help developers, administrators, and end-users set up, use, and troubleshoot the tool effectively.

The DocCentrik application is designed to:

- Search for files within a directory.
- Extract text from documents using keywords and regex patterns.
- Log matches and processing details.
- Securely upload files to an SFTP server (if enabled).
- Generate daily log files for easy monitoring.

**System Requirements**

Ensure your system meets the following requirements:

- **Operating System**: Windows 10 or later.
- **.NET Framework**: Version 4.8 or later.
- **Target Framework**: .NET 8.0.
- **Tesseract OCR Data**: Installed and configured Tesseract data files.
- **Hardware**: Minimum 8GB RAM, 100GB free disk space.
- **SFTP Server**: Valid credentials and access.

**Project Structure**

The project is organized as follows:

1. **Config Folder**:
    - **Config.cs**: Defines the configuration model, including folder paths, keywords, regex patterns, and SFTP server settings.
2. **logs Folder**:
    - Daily-generated log files for processing details.
    - Example: DocCentrikLog_2025-01-15.csv and DocCentrikMatchReport_2025-01-15.csv.
3. **Services Folder**:
    - **FileProcessor.cs**: Scans directories, extracts content, and identifies matches.
    - **PDFProcessor.cs** :Handles PDF File scanning
    - **SftpUploader.cs**: Handles file uploads to an SFTP server.

4. **Utils Folder**:
    - **Logger.cs**: Logs information into daily CSV files for processing and match results.
5. **Program.cs**:
    - The main application entry point that orchestrates file scanning, logging, and conditional SFTP uploads.
6. **config.json**:
    - A JSON file containing user-configurable settings, including a flag to enable or disable SFTP uploads.

**Installation and Setup**

**Step 1: Download and Extract**

1. Obtain the installation package from the repository or provided download link.
2. Extract the ZIP file to a location of your choice.

**Step 2: Prerequisites Installation**

1. Install the required **.NET Framework**.
2. Download and configure **Tesseract OCR Data** from the Tesseract GitHub repository.

**Step 3: Configure Settings**

1. Locate the config.json file in the extracted directory.
2. Open the file in a text editor (e.g., Notepad++ or Visual Studio Code).
3. Update the settings based on your requirements. See the **Configuration Details** section for more information.

**Step 4: Install Dependencies**

Open the NuGet Package Manager Console in your IDE and run:

dotnet add package ClosedXML --version 0.104.2

dotnet add package DocumentFormat.OpenXml --version 3.2.0

dotnet add package PdfSharpCore --version 1.3.65 dotnet

add package SSH.NET --version 2024.2.0 dotnet

add package Tesseract --version 5.2.0

**Step 5: Build and Run**

1. Build the project using Visual Studio or your preferred IDE.
2. Run the application by executing DocCentrik.exe.

**Configuration Details**

The config.json file allows you to customize the application. Below are the configurable fields:

```
{
  "FolderPath": "C:\\Users\\YourName\\Documents",
  "Keywords": ["Account", "Customer", "Loan"],
  "RegexPatterns": [
    {
      "Pattern": "[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,}",
      "Description": "Email addresses"
    }
  ],
  "SearchMode": "both",
  "SupportedExtensions": [".txt", ".log", ".csv", ".err", ".pdf", ".docx", ".xlsx", ".ppt", ".pptx"],
  "OutputLogPath": "C:\\Logs",
  "MatchReportPath": "C:\\Logs",
  "TesseractDataPath": "C:\\Tesseract\\tessdata",
  "EnableSftp": true,
  "SftpServer": {
    "Host": "sftp.example.com",
    "Port": 22,
    "Username": "username",
    "Password": "password"
  }
}
```

**Key Parameters:**

- **EnableSftp**: Set to true to enable SFTP uploads. Set to false to disable.
- **OutputLogPath** and **MatchReportPath**: Specify the directory where daily logs will be created.
- **SupportedExtensions**: Includes additional file formats such as .ppt and .pptx for comprehensive processing.

**Code Snippets**

**Daily Log Files**

The Logger.cs ensures that daily log files are generated:

```
private void InitializeLogFiles()
{
    var date = DateTime.Now.ToString("yyyy-MM-dd");
    _logFileName = Path.Combine(_logDirectory, $"DocCentrikLog_{date}.csv");
    _matchReportFileName = Path.Combine(_logDirectory, $"DocCentrikMatchReport_{date}.csv");

    if (!File.Exists(_logFileName))
    {
        File.WriteAllText(_logFileName, "Timestamp,File Name,Status\n");
```

```
    }

    if (!File.Exists(_matchReportFileName))
    {
        File.WriteAllText(_matchReportFileName, "Timestamp,File Name,Matching Word,File Type,Match Found In,OCR or Non-OCR\n");
    }
}
```

**Adding .ppt and .pptx Support**

The ExtractContent method in FileProcessor.cs includes support for .ppt and .pptx:

```
if (filePath.EndsWith(".ppt") || filePath.EndsWith(".pptx"))
{
    return ExtractFromPowerPoint(filePath);
}
```

The ExtractFromPowerPoint method:

```
private string ExtractFromPowerPoint(string filePath)
{
    using var presentation = PresentationDocument.Open(filePath, false);
    var slides = presentation.PresentationPart.SlideParts;
    return string.Join("\n", slides.SelectMany(slide => slide.Slide.Descendants<DocumentFormat.OpenXml.Drawing.Text>().Select(t => t.Text)));
}
```

**Usage Instructions**

**Step 1: Start the Application**

1. Run DocCentrik.exe.
2. The application will load the configuration and begin scanning files in the specified folder path.

**Step 2: Monitor Progress**

- The console will display real-time updates for processed files.
- Matches and errors will be logged in the daily CSV files specified in the configuration.

**Step 3: Review Logs**

- Navigate to the directory specified in OutputLogPath and MatchReportPath.
- Open daily log files (e.g., DocCentrikLog_2025-01-15.csv).

**Step 4: SFTP Uploads**

- If EnableSftp is set to true, matched files will be automatically uploaded to the configured SFTP server.
- Check the SFTP server for uploaded files.

**Troubleshooting**

**Common Issues**

1. **Configuration File Missing**:
    - Ensure config.json exists in the application directory.
2. **Invalid SFTP Credentials**:
    - Verify the Host, Username, and Password fields in config.json.
3. **OCR Extraction Fails**:
    - Confirm the Tesseract data path is valid and accessible.
4. **No Files Processed**:
    - Ensure the folder path is correct and contains files with supported extensions.
5. **Log Files Not Generated**:
    - Ensure the OutputLogPath and MatchReportPath directories are writable.

**FAQs**

1. **Can I add more keywords or regex patterns?**
    - Yes, edit the Keywords and RegexPatterns fields in config.json.
2. **What file types are supported?**
    - Supported extensions include .txt, .log, .csv, .err, .pdf, .docx, .xlsx, .ppt, .pptx.
3. **How do I disable SFTP uploads?**
    - Set EnableSftp to false in the configuration file.

**Conclusion**

The DocCentrik Centralized Document Processing Tool provides an efficient solution for scanning, processing, and managing documents. This guide equips you with the knowledge to set up, use, and troubleshoot the application effectively. For advanced configurations or support, contact the development team.

**Version History**

| Version | Date | Description |
|---|---|---|
| 1.0 | 2025-01-15 | Initial release of the technical document and end-user guide for DocCentrik. |