

■ Manual de Implantação - Sistema de Automação de Retornos CBR724

Sistema de Automação de Retornos Bancários CBR724

Organização:	Fundação Agência das Bacias PCJ
Departamento:	Tecnologia da Informação
Data:	10/10/2025
Versão:	2.0

■ Manual de Implantação - Sistema de Automação de Retornos CBR724

- > **Projeto:** Automação de Retornos Bancários CBR724
 - > **Organização:** Agência PCJ
 - > **Versão:** 1.0 - Produção
 - > **Data:** Outubro 2025
 - > **Autor:** Charles Oliveira
-

■ Índice

1. [Visão Geral do Sistema](#visão-geral)
 2. [Pré-requisitos](#pré-requisitos)
 3. [Estrutura de Pastas](#estrutura-de-pastas)
 4. [Instalação Passo a Passo](#instalação)
 5. [Configuração](#configuração)
 6. [Testes](#testes)
 7. [Produção](#produção)
 8. [Manutenção](#manutenção)
 9. [Solução de Problemas](#solução-de-problemas)
-

■ Visão Geral do Sistema {#visão-geral}

O que o sistema faz?

Automatiza o processamento de arquivos de retorno bancário no formato CBR724:

- ■ ****Monitora**** pasta de retorno no servidor
- ■ ****Processa**** arquivos CBR724 automaticamente
- ■ ****Integra**** com banco Access (dbBaixa2025.accdb)
- ■ ****Exclui**** arquivos IEDCBR automaticamente
- ■ ****Notifica**** via Windows sobre todos os eventos

- ****Reinicia**** automaticamente se cair (watchdog)

Arquitetura

```

SERVIDOR1 (Rede)  \\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno\
CBR724*.ret (arquivos de retorno)  IEDCBR*.ret (excluídos automaticamente)
  Processados\ (arquivos processados)  Erro\ (arquivos com erro)
▼
  MÁQUINA LOCAL (Monitor rodando aqui)
D:\Teste_Cobrança_Acess\AutomacaoRetorno\  monitor_retornos.py (monitor
principal)  watchdog_monitor.py (auto-restart)
notificador_windows.py (notificações)  config.ini (configurações)
*.bat (scripts de controle)

```

■ Pré-requisitos {#pré-requisitos}

Software Necessário

| Item | Versão | Onde Baixar |

|-----|-----|-----|

| **Python** | 3.13+ | <https://www.python.org/downloads/> |

| **Microsoft Access** | 2016+ | Já instalado no Windows |

| **Git** (opcional) | Qualquer | <https://git-scm.com/> |

Acesso à Rede

- Acesso de leitura/escrita: `\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\`
- Permissão para executar Python na máquina local

Hardware Mínimo

- **RAM:** 4GB
- **Disco:** 500MB livres
- **CPU:** Qualquer (baixo uso)
- **Sistema:** Windows 10/11

■ Estrutura de Pastas {#estrutura-de-pastas}

Pasta de Instalação

```
D:\Teste_Cobrança_Acess\AutomacaoRetorno\ ■ ■ ■ ■ ■ Arquivos Python (Código Principal) ■ ■ ■ ■ monitor_retornos.py # Monitor principal ■ ■ ■ ■ watchdog_monitor.py # Auto-restart ■ ■ ■ ■ notificador_windows.py # Notificações ■ ■ ■ ■ notificador_email.py # Notificações email (opcional) ■ ■ ■ ■ processador_cbr724.py # Processador CBR724 ■ ■ ■ ■ integrador_access.py # Integração Access ■ ■ ■ ■ config_manager.py # Gerenciador de config ■ ■ ■ ■ ■ Configuração ■ ■ ■ ■ config.ini # Configurações centralizadas ■ ■ ■ ■ requirements.txt # Dependências Python ■ ■ ■ ■ ■ Scripts de Controle (.bat) ■ ■ ■ ■ INICIAR_MONITOR_OCULTO.bat # Inicia monitor ■ ■ ■ ■ PARAR_MONITOR.bat # Para monitor ■ ■ ■ ■ STATUS_MONITOR.bat # Verifica status ■ ■ ■ ■ INICIAR_WATCHDOG.bat # Inicia watchdog ■ ■ ■ ■ PARAR_WATCHDOG.bat # Para watchdog ■ ■ ■ ■ PROCESSAR_EXISTENTES.bat # Reprocessa arquivos ■ ■ ■ ■ ■ Scripts PowerShell ■ ■ ■ ■ _start_monitor_hidden.ps1 ■ ■ ■ ■ _stop_all_monitors.ps1 ■ ■ ■ ■ _check_monitor.ps1 ■ ■ ■ ■ PROCESSAR_EXISTENTES.ps1 ■ ■ ■ ■ ■ Documentação ■ ■ ■ ■ MANUAL_IMPLANTACAO_COMPLETO.md # Este arquivo ■ ■ ■ ■ GUIA_CONFIG.md ■ ■ ■ ■ NOTIFICACOES_WINDOWS.md ■ ■ ■ ■ SISTEMA_WATCHDOG.md ■ ■ ■ ■ CHANGELOG.md ■ ■ ■ ■ ■ Logs (Criados automaticamente) ■ ■ ■ ■ monitor_retornos.log ■ ■ ■ ■ watchdog.log
```

Pastas no Servidor

```
\\SERVIDOR1\CobrançaPCJ\CobrançaPCJ\ ■ ■ ■ ■ Retorno\ # Pasta monitorada ■ ■ ■ ■ CBR724*.ret # Arquivos a processar ■ ■ ■ ■ Processados\ # Arquivos processados ■ ■ ■ ■ Erro\ # Arquivos com erro ■ ■ ■ ■ backup\ # Backups do Access ■ ■ ■ ■ dbBaixa2025.accdb # Banco principal ■ ■ ■ ■ Cobrança2019.accdb # Banco secundário (opcional)
```

■ Instalação Passo a Passo {#instalação}

Passo 1: Instalar Python

1. Baixe Python 3.13: <https://www.python.org/downloads/>
2. ****IMPORTANTE:**** Marque "Add Python to PATH"
3. Instale com configurações padrão
4. Verifique:

```
python --version # Deve mostrar: Python 3.13.x
```

Passo 2: Criar Estrutura de Pastas

```
# Criar pasta principal New-Item -ItemType Directory -Path
"D:\Teste_Cobrança_Acess\AutomacaoRetorno" -Force # Criar pastas no servidor (se
não existirem) New-Item -ItemType Directory -Path
"\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno\Processados" -Force New-Item -ItemType
Directory -Path "\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno\Erro" -Force New-Item
-ItemType Directory -Path "\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\backup" -Force
```

Passo 3: Copiar Arquivos do Sistema

Opção A: Via Git (Recomendado)

```
cd D:\Teste_Cobrança_Acess git clone
https://github.com/Cha-Oliveira/automacao-retorno-cbr724.git AutomacaoRetorno
```

Opção B: Cópia Manual

1. Copie todos os arquivos da pasta atual para `D:\Teste_Cobrança_Acess\AutomacaoRetorno\`
2. Mantenha a estrutura de pastas

Passo 4: Instalar Dependências Python

```
cd D:\Teste_Cobrança_Acess\AutomacaoRetorno pip install -r requirements.txt
```

Bibliotecas instaladas:

- `watchdog==3.0.0` - Monitoramento de arquivos
 - `pyodbc==4.0.39` - Conexão com Access
 - `pyyaml==6.0.1` - Configurações
 - `colorama==0.4.6` - Cores no console
 - `psutil==5.9.8` - Monitoramento de processos
 - `plyer==2.1.0` - Notificações Windows
-

■■ Configuração {#configuração}

Arquivo config.ini

Edite o arquivo `config.ini` com as configurações corretas:

```
[CAMINHOS] # Pasta monitorada no servidor pasta_retorno =
\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno # Pasta para arquivos processados
pasta_processados = \\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno\Processados # Pasta
para arquivos com erro pasta_erro =
\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno\Erro # Pasta para backups pasta_backup
= \\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\backup [BANCOS_ACCESS] # Banco principal
db_baixa = \\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\dbBaixa2025.accdb # Banco secundário
```

```
(opcional) db_cobranca = \\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Cobranca2019.acddb
usar_cobranca = false [PYTHON] # Caminho do Python (ajuste se necessário)
executavel =
C:\Users\SEU_USUARIO\AppData\Local\Programs\Python\Python313\python.exe [LOGS]
arquivo_log = monitor_retornos.log nivel_log = INFO [PROCESSAMENTO]
tempo_espera_arquivo = 1 fazer_backup = true excluir_ied = true [NOTIFICACOES] #
Notificações do Windows habilitado = true [EMAIL] # Notificações por email
(opcional - pode deixar desabilitado) habilitado = false smtp_servidor =
smtp.gmail.com smtp_porta = 587 remetente = seuemail@agencia.baciaspcj.org.br senha
= sua_senha_de_app destinatarios = destinatario@agencia.baciaspcj.org.br
```

Ajustar Caminho do Python

Para descobrir o caminho correto do Python:

```
Get-Command python | Select-Object -ExpandProperty Source
```

Copie o resultado e cole em `config.ini` na seção `[PYTHON]`.

■ Testes {#testes}

Teste 1: Verificar Configuração

```
cd D:\Teste_Cobrança_Acess\AutomacaoRetorno python config_manager.py
```

Resultado esperado: Mostra todas as configurações sem erros.

Teste 2: Testar Notificações

```
python notificador_windows.py
```

Resultado esperado: Notificação aparece no canto da tela.

Teste 3: Iniciar Monitor (Teste Rápido)

```
python monitor_retornos.py
```

Resultado esperado:

- Mensagem "Monitor iniciado"
- Notificação do Windows aparece
- Aguardando arquivos...

Pressione `Ctrl+C` para parar.

Teste 4: Processar Arquivo de Teste

1. Copie um arquivo CBR724 para a pasta monitorada:

```
Copy-Item "caminho\do\arquivo\teste.ret" -Destination  
"\\SERVIDOR1\CobrancaPCJ\CobrancaPCJ\Retorno\"
```

2. Observe:

- Notificação "Arquivo Detectado"
 - Log mostra processamento
 - Notificação "Arquivo Processado" (se sucesso)
 - Arquivo movido para `Processados`
-

■ Produção {#produção}

Iniciar Sistema Completo

1. Inicie o Monitor:

```
INICIAR_MONITOR_OCULTO.bat
```

2. Inicie o Watchdog (Opcional mas Recomendado):

```
INICIAR_WATCHDOG.bat
```

Verificar Status

```
STATUS_MONITOR.bat
```

Parar Sistema

```
PARAR_MONITOR.bat PARAR_WATCHDOG.bat
```

Configurar Início Automático com Windows

Opção 1: Agendador de Tarefas (Recomendado)

1. Abra "Agendador de Tarefas" (`taskschd.msc`)
2. Criar Tarefa Básica
3. Nome: "Monitor Retornos CBR724"
4. Gatilho: "Ao iniciar"
5. Ação: Iniciar programa
 - Programa: `D:\Teste_Cobrança_Acess\AutomacaoRetorno\INICIAR_MONITOR_OCULTO.bat`
6. Marcar: "Executar com privilégios mais altos"

Repita para o watchdog se desejar.

Opção 2: Pasta de Inicialização

1. `Win + R` → `shell:startup`
 2. Criar atalho de `INICIAR_MONITOR_OCULTO.bat`
 3. Criar atalho de `INICIAR_WATCHDOG.bat`
-

■ Manutenção {#manutenção}

Verificar Logs

Monitor:

```
Get-Content "D:\Teste_Cobrança_Acess\AutomacaoRetorno\monitor_retornos.log" -Head 20
```

Watchdog:

```
Get-Content "D:\Teste_Cobrança_Acess\AutomacaoRetorno\watchdog.log" -Head 20
```

Reprocessar Arquivos Existentes

```
PROCESSAR_EXISTENTES.bat
```

Este script:

1. Exclui arquivos IEDCBR
2. Move arquivos CBR de volta para reprocessamento

Atualizar Sistema

Se houver nova versão:

```
cd D:\Teste_Cobrança_Acess\AutomacaoRetorno # Parar sistema PARAR_MONITOR.bat  
PARAR_WATCHDOG.bat # Atualizar via Git git pull # Reinstalar dependências (se  
mudaram) pip install -r requirements.txt --upgrade # Reiniciar  
INICIAR_MONITOR_OCULTO.bat INICIAR_WATCHDOG.bat
```

Backup da Configuração

```
# Copiar config.ini para backup Copy-Item "config.ini" -Destination  
"config.ini.backup" # Copiar para OneDrive Copy-Item "config.ini" -Destination  
"C:\Users\SEU_USUARIO\OneDrive\Manuais\AutomacaoRetorno\"
```

■■ Solução de Problemas {#solução-de-problemas}

Problema: Monitor não inicia

Erro: "can't open file monitor_retornos.py"

Solução:

1. Verifique se está na pasta correta
2. Execute:

```
cd D:\Teste_Cobrança_Acess\AutomacaoRetorno python monitor_retornos.py
```

Problema: Erro de conexão com Access

Erro: "Não foi possível conectar ao banco de dados"

Causas possíveis:

1. Banco está aberto em outra máquina
2. Caminho incorreto no `config.ini`
3. Permissões insuficientes

Solução:

1. Feche o Access em outras máquinas
2. Verifique o caminho em `config.ini`
3. Teste acesso manual ao banco

Problema: Notificações não aparecem

Solução:

1. Verifique se `plyer` está instalado:

```
pip list | Select-String plyer
```

2. Instale se necessário:

```
pip install plyer
```

3. Desative "Foco Assistente" do Windows

Problema: Watchdog não reinicia monitor

Erro: "ModuleNotFoundError: No module named 'psutil'"

Solução:

```
pip install psutil
```

Problema: Arquivo não é processado

Verificar:

1. Arquivo está na pasta correta?
2. Extensão é `.ret`?
3. Arquivo é IEDCBR? (excluído automaticamente)
4. Monitor está rodando?
5. Verifique logs

Problema: Muitos processos Python rodando

Solução:

```
# Para todos os monitores PARAR_MONITOR.bat # Ou manualmente Get-Process python |  
Where-Object {$_.CommandLine -like "*monitor_retornos*"} | Stop-Process -Force
```

■ Monitoramento

KPIs para Acompanhar

Métrica	Como Verificar
Monitor está rodando?	`STATUS_MONITOR.bat`
Últimos arquivos processados	Pasta `Processados`
Erros recentes	Pasta `Erro` + logs
Uptime do sistema	`watchdog.log`
Notificações enviadas	Central de Ações Windows

Frequência de Verificação

- **Diária:** Verificar se monitor está rodando
 - **Semanal:** Revisar logs de erro
 - **Mensal:** Atualizar dependências Python
-

■ Suporte

Contatos

- **Desenvolvedor:** Charles Oliveira
- **Email:** charles.oliveira@agencia.baciaspcj.org.br
- **Organização:** Agência PCJ

Documentação Adicional

- `GUIA_CONFIG.md` - Configurações detalhadas
 - `NOTIFICACOES_WINDOWS.md` - Sistema de notificações
 - `SISTEMA_WATCHDOG.md` - Auto-restart
 - `CHANGELOG.md` - Histórico de mudanças
-

■ Checklist de Implantação

Use este checklist ao implantar em nova máquina:

- [] Python 3.13+ instalado
 - [] Pasta criada: `D:\Teste_Cobrança_Acess\AutomacaoRetorno\`
 - [] Arquivos copiados
 - [] Dependências instaladas (`pip install -r requirements.txt`)
 - [] `config.ini` configurado corretamente
 - [] Caminho do Python ajustado
 - [] Teste de notificações OK
 - [] Teste de processamento OK
 - [] Monitor iniciado em produção
 - [] Watchdog iniciado (opcional)
 - [] Início automático configurado
 - [] Backup do `config.ini` feito
-

■ Notas Finais

Boas Práticas

1. ■ Sempre use o watchdog em produção
2. ■ Revise logs semanalmente
3. ■ Faça backup do `config.ini`
4. ■ Teste antes de atualizar
5. ■ Mantenha documentação atualizada

Segurança

- ■ Não compartilhe `config.ini` (contém caminhos sensíveis)
 - ■ Não use senhas de email normal (use senha de app)
 - ■ Mantenha backups do sistema
-

■ Sistema pronto para produção!

Data de criação deste manual: Outubro 2025

Versão do sistema: 1.0

Status: ■ Produção