

Documentação Profissional do Projeto de Análise de Dados

Projeto: **Movimentações Financeiras**

Autor: José Augusto

Data: 31/07/2025

1. Documentação de Requisitos

Objetivo: Entender profundamente as necessidades do negócio e definir os objetivos da análise.

1.1 Levantamento de Requisitos

O case tem como foco a análise de movimentações financeiras de clientes. As principais perguntas de negócio são:

- Qual o volume total de movimentações?
- Quais clientes são mais ativos?
- Como os valores se comportam ao longo do tempo?

1.2 Especificação de Métricas e KPIs

- Total de Entradas Financeiras
- Total de Saídas Financeiras
- Saldo por Cliente
- Participação do PIX nas transações
- Distribuição mensal e semanal

1.3 Regras de Negócio

- CPF deve ser padronizado e anonimizado
- Entradas e saídas são registradas separadamente
- Datas devem estar formatadas como YYYY-MM-DD

2. Documentação Técnica

Objetivo: Descrever como os dados são coletados, tratados e armazenados.

2.1 Dicionário de Dados (Exemplo)

Tabela	Coluna	Tipo	Descrição
d_clientes	cpf	string	CPF anonimizado
f_entrada_fluxo	transaction_id	int	ID da transação de entrada
f_saida_fluxo	amount	float	Valor da saída
d_contas	created_at	timestamp	Data de criação da conta

2.2 Modelo de Dados (ERD)

- Tabelas dimensão: d_clientes, d_contas, d_cidade, d_estado, d_pais
 - Tabelas fato: f_entrada_fluxo, f_saida_fluxo, f_mov_pix
- Relacionamentos definidos via chaves: id_cliente, id_conta, id_cidade, etc.

2.3 Pipeline de Dados

- Ingestão: leitura de arquivos .csv
- Transformação: tratamento de CPF, datas, timestamps, duplicatas e nulos
- Carga: inserção no PostgreSQL com log e auditoria

2.4 Scripts

- Linguagem: Python
- Bibliotecas: pandas, sqlalchemy, logging, psycopg2
- Estrutura modular com funções para validação, limpeza e log

2.5 Configuração de Conexões

Arquivo db_config.json contém as credenciais para o PostgreSQL.

Formato esperado:

```
{  
  
  "user": "usuario",
```

```
"password": "senha",  
"host": "localhost",  
"port": "5432",  
"database": "nome_banco"  
}
```

3. Documentação de Processos

Objetivo: Garantir a replicabilidade e manutenção da pipeline.

3.1 Manual de Execução

- Executar via terminal: `python pipelinetestefinal.py`
- Automatizar via `.bat` + Agendador de Tarefas

3.2 Arquivo .BAT (Exemplo)

```
@echo off
```

```
cd C:\caminho\do\projeto
```

```
python pipelinetestefinal.py
```

```
pause
```

3.3 Orquestração

- Utilização do Agendador do Windows
- Alternativas viáveis: Pentaho ou Apache Airflow para pipelines mais robustos

4. Documentação Analítica

Objetivo: Justificar decisões com base na análise dos dados transformados.

4.1 Metodologia (EDA)

- Análise inicial com `pandas` e `seaborn`
- Visualização final em Power BI

- Verificação de outliers, sazonalidade e agrupamentos

4.2 Hipóteses Testadas

- Transações se concentram no início de cada mês
- Contas mais antigas movimentam valores maiores
- O uso do PIX vem crescendo mês a mês

4.3 Visualizações e Dashboards (Power BI)

Tipo de Layout

- Dashboard financeiro com blocos bem definidos
- KPIs no topo, filtros laterais e gráficos no corpo central

Paleta de Cores

- Azul primário (#007ACC)
- Verde para entradas, vermelho para saídas
- Tons de cinza para fundo e texto secundário

Carga Incremental

- Implementada com base na data da última atualização
- Colunas como data_entrada, data_saida, data_mov_pix usadas como referência
- Evita recarga completa a cada atualização

Boas Práticas

- Tabela de medidas centralizada
- Nomes claros e padronizados
- Uso de filtros visuais e hierárquicos
- Responsividade ajustada para múltiplos dispositivos

4.4 Medidas DAX (Exemplos)

Total Entradas = SUM(f_entrada_fluxo[amount])

Total Saídas = SUM(f_saida_fluxo[amount])

Saldo = [Total Entradas] - [Total Saídas]

% Contas Ativas =

DIVIDE(

```
CALCULATE(COUNTROWS(d_contas), d_contas[status] = "Ativo"),  
COUNTROWS(d_contas),  
0  
)
```

4.5 Compartilhamento

- Publicação no Power BI Service
- Atualização automática com Gateway local
- Segurança via RLS (se necessário)

5. Documentação de Qualidade e Governança

Objetivo: Garantir a confiabilidade e rastreabilidade dos dados.

5.1 Controle de Versões

- Histórico dos arquivos .csv salvo localmente
- Versionamento de código com Git

5.2 Logs e Auditoria

- Log salvo em pipeline2.log
- Cada execução registra data/hora, erros, colunas novas e volume de dados

5.3 Tratamento de Dados Sensíveis

- CPF anonimizado e convertido para string de 11 dígitos
- Dados pessoais nunca expostos nos dashboards

5.4 SLA de Atualização

- Atualização semanal automática
- SLA de resposta: 24 horas em caso de falha

6. Documentação de Projetos

Objetivo: Monitorar entregas e evolução do projeto.

6.1 Cronograma de Entregas

Etapa	Início	Fim
Requisitos e Planejamento	01/07/2025	03/07/2025
Desenvolvimento Pipeline	04/07/2025	15/07/2025
Validação e Testes	16/07/2025	25/07/2025
Publicação Power BI	26/07/2025	28/07/2025

6.2 Gestão de Tarefas

- Utilização do **Trello** para controle de tarefas
- **Confluence** para centralizar a documentação técnica da equipe
- **GitHub** para versionamento do pipeline e dos scripts

6.3 Lições Aprendidas

- Automatizar com logs facilita suporte e auditoria
- O uso de carga incremental reduz significativamente o tempo de atualização
- Importância da padronização dos dados de entrada