

ESBOÇO DE IMPLEMENTAÇÃO RUST – GOODWARE

Autor: Wolfspell & Collaborative AIs | Idioma: pt | Data: 2025-07-06

```
use analytics_crate;

use pdf_report_generator;

// Coletor de dados
struct DataIngestor;

impl DataIngestor {
    fn collect_data(&self) -> Data {
        println!("Coletando dados de produção...");

        Data {}
    }
}

// Analisador de emissões
struct EmissionAnalyzer;

impl EmissionAnalyzer {
    fn analyze(&self, data: &Data) -> AnalysisResult {
        println!("Analisando emissões...");

        AnalysisResult {}
    }
}

// Otimizador de processo
struct ProcessOptimizer;

impl ProcessOptimizer {
    fn optimize(&self, analysis: &AnalysisResult) -> Recommendations {
        println!("Gerando recomendações...");

        Recommendations {}
    }
}
```

```
    }  
}  
  
struct Data {}  
struct AnalysisResult {}  
struct Recommendations {}  
  
fn main() {  
    println!("Verificando segurança e conformidade...");  
  
    let ingestor = DataIngestor;  
    let analyzer = EmissionAnalyzer;  
    let optimizer = ProcessOptimizer;  
  
    let data = ingestor.collect_data();  
    let analysis = analyzer.analyze(&data);  
    let recommendations = optimizer.optimize(&analysis);  
  
    println!("Gerando relatório PDF semanal...");  
    // pdf_report_generator::generate(recommendations, "report.pdf");  
}
```

Este código é um esboço; deve ser auditado por especialistas antes de uso operacional. Usa tipagem forte, evita 'unsafe' e será integrado a logs imutáveis e quorum ético conforme o manifesto.