

# ESQUISSE D'IMPLÉMENTATION RUST – GOODWARE

Auteur : Wolfspell & Collaborative AIs | Langue : fr | Date : 2025-07-06

```
use analytics_crate;
use pdf_report_generator;

// Collecteur de données
struct DataIngestor;
impl DataIngestor {
    fn collect_data(&self) -> Data {
        println!("Collecte des données de production...");
        Data {}
    }
}

// Analyseur d'émissions
struct EmissionAnalyzer;
impl EmissionAnalyzer {
    fn analyze(&self, data: &Data) -> AnalysisResult {
        println!("Analyse des émissions...");
        AnalysisResult {}
    }
}

// Optimiseur de processus
struct ProcessOptimizer;
impl ProcessOptimizer {
    fn optimize(&self, analysis: &AnalysisResult) -> Recommendations {
        println!("Génération des recommandations...");
        Recommendations {}
    }
}

struct Data {}
struct AnalysisResult {}
struct Recommendations {}
```

```
fn main() {  
    println!("Vérification sécurité et conformité...");  
    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!("Génération du rapport PDF hebdomadaire...");  
    // pdf_report_generator::generate(recommendations, "rapport.pdf");  
}
```

*Ce code est un prototype■; un audit d'experts est requis avant toute mise en production. Typage fort, sans 'unsafe', intégration à logs immuables et quorum éthique selon le manifeste.*