GOODWARE RUST IMPLEMENTATION – Sketch

Author: Wolfspell & Collaborative AIs | Language: en | Date: 2025-07-06

```
use analytics_crate;
use pdf_report_generator;

// Data collector
struct DataIngestor;
impl DataIngestor {
    fn collect_data(&self) -> Data {
        println!("Collecting production data...");
        Data {}
    }
}

// Emission analyzer
struct EmissionAnalyzer;
impl EmissionAnalyzer {
    fn analyze(&self, data: &Data) -> AnalysisResult {
        println!("Analyzing emissions...");
        AnalysisResult {}
    }
}

// Process optimizer
struct ProcessOptimizer;
impl ProcessOptimizer {
    fn optimize(&self, analysis: &AnalysisResult) -> Recommendations {
        println!("Generating recommendations...");
        Recommendations {}
    }
}

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

```
fn main() {
    println!("Checking safety and compliance...");
    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!("Generating weekly PDF report...");
    // pdf_report_generator::generate(recommendations, "report.pdf");
}
```

This code is a sketch; it must be audited by specialists before operational use. It relies on strong typing, avoids 'unsafe', and will integrate immutable logs and ethical quorum according to the manifesto.