

# RUST■IMPLEMENTIERUNGSSKIZZE – GOODWARE

Autor: Wolfspell & Collaborative Als | Sprache: de | Datum: 2025-07-06

```
use analytics_crate;
use pdf_report_generator;

// Daten■Sammler
struct DataIngestor;
impl DataIngestor {
    fn collect_data(&self) -> Data {
        println!("Produktionsdaten werden gesammelt...");
        Data {}
    }
}

// Emissions■Analysator
struct EmissionAnalyzer;
impl EmissionAnalyzer {
    fn analyze(&self, data: &Data) -> AnalysisResult {
        println!("Emissionen werden analysiert...");
        AnalysisResult {}
    }
}

// Prozess■Optimierer
struct ProcessOptimizer;
impl ProcessOptimizer {
    fn optimize(&self, analysis: &AnalysisResult) -> Recommendations {
        println!("Optimierungsempfehlungen werden erstellt...");
        Recommendations {}
    }
}

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

```
fn main() {  
    println!("Sicherheits■ und Compliance■Check...");  
    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!("Wöchentlicher PDF■Bericht wird erzeugt...");  
    // pdf_report_generator::generate(recommendations, "bericht.pdf");  
}
```

*Dies ist eine Implementierungsskizze; vor Produktionseinsatz ist eine Expertenprüfung erforderlich. Starker Typ■Check, kein 'unsafe', Integration in unveränderliche Logs und ethisches Quorum gemäß Manifest.*