

Swift Code Review: iPadScannerApp Source

1. RT60Evaluator.swift

- **Positiv:** Klare Bewertungslogik gem DIN 18041
- **Empfehlung:** Toleranz (15 %) als Konstante deklarieren
- **Verbesserung:** Logging-Optionen + gewichtete Frequenzbewertung

2. AbsorberCalculator.swift

- **Positiv:** Nutzung der Sabine-Formel, saubere Struktur
- **Empfehlung:** Schutz vor Division durch 0, R ckgabe via `Result`
- **Verbesserung:** Bessere Dokumentation der Berechnungsgrundlagen

3. RT60ChartView.swift

- **Positiv:** SwiftUI + Charts-Integration gelungen
- **Empfehlung:** Achsenbeschriftung erg nzen
- **Verbesserung:** Optional `FrequencyBand` als enum zur Strukturierung

4. USBMicIntegration.swift

- **Positiv:** AVFoundation-Integration sauber gel st
- **Empfehlung:** Fehlerhandling statt `try?`, R ckgabewert bei Auswahl
- **Verbesserung:** UI-R ckmeldung via Notification/Delegate

Zusammenfassung

- Sehr gute Modularisierung
- Fehlerbehandlung und Doku ausbauf hig
- Testbarkeit gegeben, aber f r AV/Chart eingeschr nkt
- Empfehlung: LoggingService, UnitTests, DocC-Kommentierung