

**P-Ai**

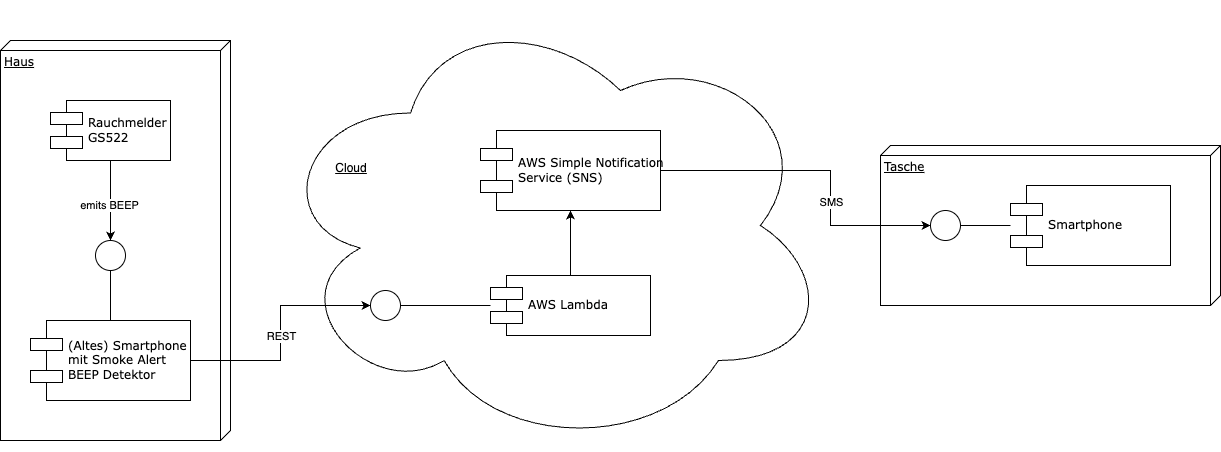
**GVG Smoke Voice**

**Technische Informationen für die Jury**

**Aktueller Stand des Sourcecodes**<https://github.com/tjahn/piaapp>

**Ausgangslage**

Aufgrund des sehr beschränkten Zeitrahmens haben wir konsequent das YAGNI-Prinzip angewendet. YAGNI ist ein Akronym und steht für You Ain't Gonna Need It.  
Wir implementierten für unseren MVP nur wirklich benötigte Features.

**Technischer Aufbau**

* Smoke Alert BEEP Detektor: Mobile App geschrieben in **Flutter.** Flutter ist ein plattformübergreifendes UI-Toolkit, das es uns ermöglicht unseren Code über verschiedene Betriebssysteme hinweg zu verwenden (iOS und Android).
* Cloud: **AWS** Lambda Funktionen zum Bereitstellen eines REST-Endpunktes sowie **Amazon SNS**, um SMS-Nachrichten direkt an eine Telefonnummer zu senden.

**Implementation**

Die Herausforderung war mit Hilfe eines FFT-Filters (Fast-Fourier-Transformation) das Piepsen des Rauchmelders zu detektieren sowie dieses aus den Umgebungsgeräuschen herauszufiltern und zu entzerren.  
  
Die Mobile App muss zuerst auf des emitierte Biepen des Rauchmelders geeicht werden. Dies wird durch das Betätigen des Knopfes “Take sample” erreicht.

Danach ist die Mobile App in der Lage das Biepen zuverlässig aus Umgebungsgeräuschen herauszufiltern. Wird ein vordefinierte Schwelle überschritten, sendet die Mobile App einen Request an unsere AWS Backendlösung welche ein SMS-Nachricht an die hinterlegte Telefonnummer auslöst.  
  
Dies Herausfiltern des Piepsignals ist aus technischer und algorithmischer Sicht das besonders Coole an unserer Lösung.

**Abgrenzung / Offene Punkte**

Keine

