

BärnHäckt – Feuerlöscher

Mitglieder Alessio Petrone
Benedikt Dalferth
Christian Ottenhaus
Jan Koch

Challenge Smoke Voice

Ausgangslage

Rauchmelder sind bisher oftmals analog und warnen mit ihrem Alarmton nur Personen im Haus oder direkt davor. Die GVB möchte ihren Kunden ermöglichen, auch unterwegs benachrichtigt zu werden, wenn ihr Rauchmelder einen Alarm meldet, um Brände schneller bekämpfen zu können. Smart Speaker wie Apple Home oder Alexa bieten diese Möglichkeit bereits an, allerdings zulasten des Datenschutzes.

Ebenfalls gibt es auf dem Markt smarte Rauchmelder, die eine Benachrichtigungsfunktion direkt integriert haben. Für die Nachhaltigkeit und Langlebigkeit von nicht-smarten Bestandsrauchmeldern soll eine einfache und kostengünstige Alternative zu Smart Speakern geschaffen werden.

Lösung

Hierfür haben wir ein smartes Device entwickelt, welches den Alarmton des Rauchmelders erkennt und via WLAN eine Nachricht an den GVB Kunden versendet. Dabei erfolgt die Erkennung des Alarmsignals über ein speziell dafür trainiertes **KI-Model** lokal auf dem smarten Device, um die Erkennung möglichst **präzise und sicher** zu gestalten. Die Audiodaten bleiben dabei auf dem Gerät und gelangen nicht ins Internet.

In der an den Kunden versendete Nachricht ist ein Link zur GVB Plattform. Dort kann der Kunde einen **Video-Livestream** aus dem Bereich sehen, in dem der Rauch erkannt wurde. So kann er überprüfen, ob wirklich ein Brand vorherrscht. Im Notfall kann die **Feuerwehr einfach** per Knopfdruck **alarmiert** werden. So wird das Feuer schnellstmöglich gelöscht, auch von der anderen Seite der Welt, **sicher und datenschutzfreundlich**.

Technischer Aufbau & Implementation

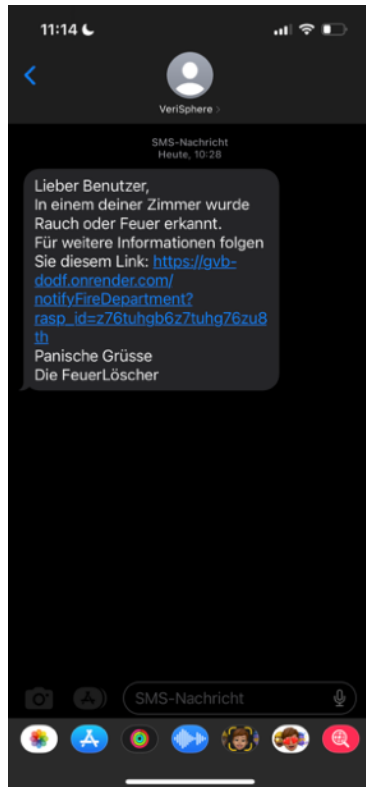
Die Alarmton-Erkennung funktioniert mit einem speziell dafür trainierten KI-Modell lokal auf einem Raspberry Pi. Das KI-Modell basiert auf Google Tensorflow. Umgesetzt wurde die Programmierung mit Python. Sobald ein Alarmton erkannt wird, sendet der Raspberry Pi eine Benachrichtigung an unsere Plattform, welcher alle für das Haus angemeldeten Nutzer per SMS und/oder E-Mail benachrichtigt. Die Plattform basiert dabei auf dem MERN-Stack (MongoDB, Express.js, React und Node.js).

Um zu prüfen, dass der Alarm kein Fehlalarm ist, kann der Nutzer einen Livestream von der am Raspberry Pi angebrachten Kamera auf der Benachrichtigungs-Website sehen. Dies ist mit Websocket implementiert.

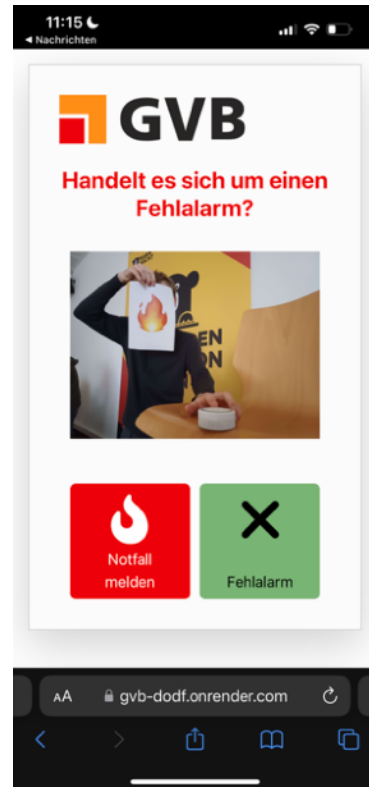
Weitere Features

- **Schnellere Brandlöschung:** Direkte Alarmierung der Feuerwehr bei Alarm mit Kundendaten wie Grundrisse, exakter Brandort etc.
- **Weitere Benachrichtigungs-Integrationen:** z.B.: Push Notification, WhatsApp, Signal, Threema etc.
- **Kosten stark senken:** Raspberry Pi durch deutlich günstigeren Micro-Controller ersetzen
- **Flexibilität und Unabhängigkeit:** Akkueinsatz und Einbindung ins Mobilfunk-Netz (z.b. dezentrale Netze wie Helium), um Abseits von Steckdosen und WLAN zu funktionieren
- **Weitere Smart Home Integrationen:** Apple HomeKit, Amazon Alexa, Google Home
- **Weitere Sensoren:** Neben Rauchmeldern auch Wassersensoren oder Windsensoren etc.

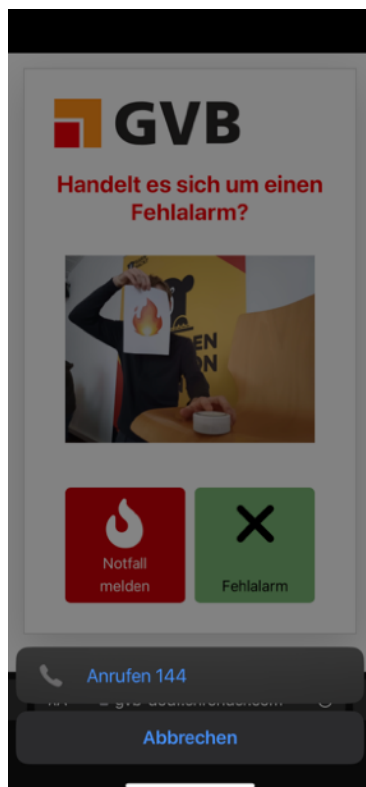
Screenshots



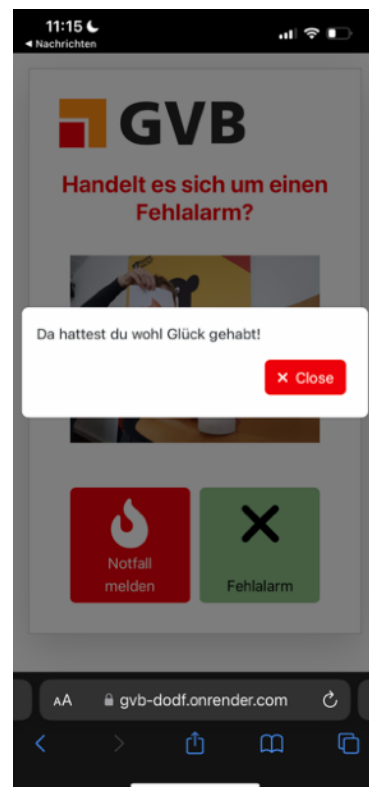
Bei erkanntem Rauch
wird SMS versendet



Video-Feed zum schauen,
ob es ein Fehlalarm ist



Notfall melden
→ Direkt 144 anrufen



Fehlalarm
→ Nochmal Glück gehabt

Links

Pitch-Anmeldung	https://hackme.qrcheck.app
Brand-Validierung	https://hackme.qrcheck.app/notification
Raspberry - Github	https://github.com/QRCheckApp/bernhackt23-raspberrypi
WebApp - Github	https://github.com/JanKoch99/Feuerloescher