

*Die Zukunft der Heimautomation
schon heute erleben*

Keyless Entry

von Thomas Bohn, Sven Ehmer, Timo Matuszewski, Fabian Künne, Jan Rathmer

Inhaltsverzeichnis

- Projektziele
 - Idee & Architektur
- Smartphone
- Serverfunktionen
- Türfalle
 - Technische Besonderheiten
- Demo
- Probleme
- Projektziele & Ausblick
- Fazit

Projektziele

- Schlüssellos Öffnen einer Tür
- Bedienung einfach halten
- Nutzerinteraktion minimieren
- Stromsparend

Idee

- Authentifizierung mittels Smartphone
 - Authentifizierung im Hintergrund
 - Energiesparendes Bluetooth LE
 - Soll auch Offline möglich sein
- Raspberry Pi
 - Nimmt Authentifizierung entgegen
 - Steuert elektronischen Türöffner

Architektur

- Smartphone als Client
 - Smartphone-App (Android)
 - App als Backgroundprozess
- Raspberry Pi als Server
 - Node.js
 - MongoDB
 - Bleno/BlueZ
- Türfalle
 - Elektrische Türfalle
 - Darlington-Schaltung zum Ansteuern

Smartphone

- Android 4.3+
- Authentifizierung gegenüber Tür
- Verwalten von Türen

To-do:

- Alternative zum Webinterface
- Authentifizieren im Hintergrund
- Verwalten von Türen erweitern

Serverfunktionen

- Hauptprogramm: Node.js
 - JavaScript
 - Asynchrones Verhalten
 - Ereignisgesteuerte Architektur (Callbacks)
 - Größtenteils non-blocking
- Datenhaltung: MongoDB
 - Dokumentenorientierte NoSQL-Datenbank
 - Mit Mongoose "Objektrelationale Abbildung" in Node.js
- Bluetoothkommunikation: Bleno/BlueZ
 - BlueZ: offizieller Bluetooth-Stack unter Linux
 - Bleno: Node-Modul für Bluetooth-Zugriff

Serverfunktionen

- Webserver: ExpressJS
 - Node.js
 - Liefert Verwaltungsseite aus
 - Generiert QR-Code
 - Ruft Daten ab
- Webinterface: AngularJS
 - Open-Source MVC Framework
 - Manipuliert DOM in Echtzeit
 - Asynchrone Aufrufe
- Cryptobibliothek
 - Zeitabhängiges "One-Time-Password" (OTP)

Türfalle

- Spannungsbereich: 9 - 16 V
- Stromaufnahme: 0,48 A bei 12 V
- Gleich- und Wechselstrom
- Ansteuerung via
Darlingtonschaltung
- Abschaltung nach 60 sec.



Technische Besonderheiten

- Offline Authentifizierung
- Non-Blocking
 - Website & Bluetoothanfragen gleichzeitig
- Sicherheit durch OTA
- AngularJS

Demo

Probleme

- Android
 - Batteriedauer
 - Bluetooth LE
 - Backgroundprozess
- Raspberry Pi
 - Funktionsweise von Node.js
 - BlueZ-Versionen

Projektziele

- Schlüssellos Öffnen einer Tür ✓
- Bedienung einfach halten ✓
- Nutzerinteraktion minimieren ●
- Stromsparend ✗

Ausblick

Mögliche Erweiterungen sind

- Tür erhält weiteren Sensor “Ist der Nutzer innerhalb des Hauses?”
- Tür sendet Nachricht “Nutzer XY betritt das Haus”
- Tür lässt sich mit anderen Smart-Devices (Schlüsselanhänger, Finger-Ring, etc.) öffnen
- Tür erkennt auch RFID-Chips
- Tür verbindet sich mit Hausautomation

Fazit

- Proof-of-Concept
- Komfort deutlich gesteigert
- Funktion gut, aber:
 - Sehr energieintensiv

