IoT Projekt : Entwicklung einer Android App zur Visualisierung von emulierten Sensordaten durch einen cloud Environenment.

Dieser Dokumentation beschreibt eine Fallstudie zum Erstellen eines Smart Home mit Raspberry Pi-Gerät und die in einen Cloud Umgebung integriet ist . Wir haben AndroidThings als Betriebssystem auf dem Raspberry-Gerät und Firebase eine bekannte Service von Cloud Anbieter Google Cloud Platform (GCP) als Backend-Dienst zum Speichern der Daten verwendet (der in Echtzeit mit allen authentifizierten Geräten synchronisiert wird). Der Projekt gibt eine kurze Beschreibung des aktuellen Trends bei IoT-Geräten (populäre IoT-Plattformen wie Raspberry Pi, Arduino usw.). Der Artikel erwähnt auch eine kurze Einführung in AndroidThings OS (von Google bereitgestellt), das für IoT-Geräte entwickelt wurde und das Java-Framework für Anwendungsentwickler unterstützt, um IoT-Anwendungen mit Java zu entwickeln. Wir sprechen auch über Firebase von GCP , das als Backend zum Speichern von Daten verwendet wurde.

## AndroidThings

AndroidThings bringt jetzt die gesamte Android-Plattform auf Geräte, auf denen Sie Java-basierte IoT-Anwendungen erstellen können. Es hat das Potenzial, das Spiel für IoT zu verändern, wie es Android für Geräte getan hat.  
Das Problem, mit dem wir die meisten IoT-Entwickler konfrontiert haben, ist das Sammeln von Sensordaten, den Transport und das Speichern dieser Daten im Backend. Und der wahre Wert liegt in der Analyse dieser Daten für Alerts, Visualisierung usw. Und genau hier sehen wir viele Menschen, die die Werte von Firebase, Google Cloud Services, besser verstehen. AndroidThings, entwickelt von Google, fügt all diese Teile zu einer überzeugenden IoT-Plattform zusammen.  
Eines der interessanten Dinge, die als Stärke von Android Things genannt werden, ist, dass OS-Updates von Google selbst verteilt werden.

* **Firebase Backend as a Service von Google Cloud Platform (GCP)**

Firebase bietet eine schnelle Möglichkeit, sensorische Daten, die auf Geräteebene gesammelt wurden, dauerhaft zu speichern, und es funktioniert hervorragend mit den Android-APIs, die von Android Things unterstützt werden. Viele Programmierer von Mobilgeräten und Geräten, mit denen ich konfrontiert bin, haben Probleme mit der serverseitigen Programmierung. Firebase kann wirklich helfen, diese Lücke zu überbrücken und es einfacher zu machen.  
Es wird interessant sein zu sehen, wie Entwickler ihre Offline-Funktionen nutzen. Wenn Sie neu bei IoT sind oder allgemein jedes Gerät, das Daten sammelt und es über Netzwerke übertragen muss, ist die goldene Regel, dass keine Netzwerkverbindung angenommen werden kann. Daher müssen Sie die Daten offline sammeln und wenn das Netzwerk verfügbar ist, übertragen Sie dies auf Ihren Server. Firebase mit seiner Offline-Funktion kann das für viele Entwickler wirklich vereinfachen.  
Firebase hat eine Menge Funktionen wie Real-Time-Datenbank, Authentifizierung, Cloud Messaging, Storage, Hosting, Testlabor und Analytics, aber ich werde nur Authentifizierung, Real-Time-Datenbank verwenden.

* **Authentifizierung**

Jede geeignete App hat irgendeine Form von Sicherheit. Die Firebase-Authentifizierung bietet diese Form der Sicherheit, indem sie eine OAuth-Plattform bereitstellt, die gängige OAuth-Anbieter wie Facebook, Google+ und Twitter integriert.

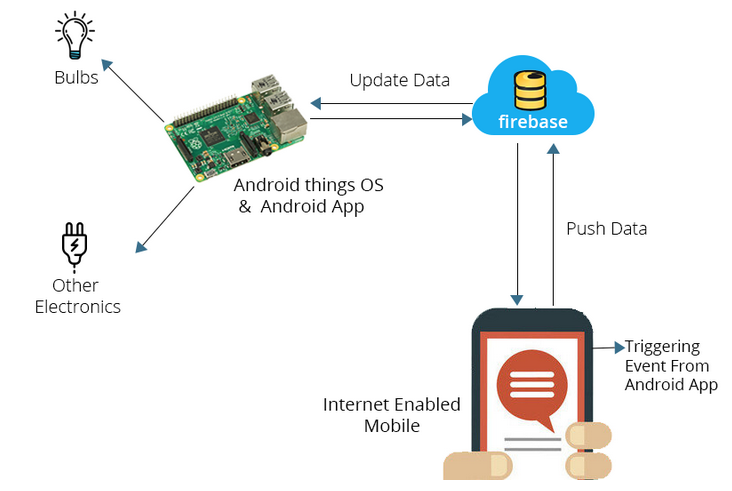
* **Echtzeitdatenbank**

Dies ist eine NoSQL-Cloud-Datenbank. Okay, das bedeutet, dass alle Daten in Ihrer Anwendung online in der Cloud gespeichert werden und ein zusätzlicher Vorteil ist, dass es in Echtzeit (wie es sich ändert) über alle verbundenen Clients synchronisiert wird.

* **Umsetzung: Smart Home mit RaspberryPi.**  
    
  Diese Implementierung erfordert Folgendes:  
    
  **Software:**  
    
  1. Java.  
  2.Android Anwendungsentwicklung.  
    
  **Hardware:**  
    
  1. HimbeerePi 3  
  2. Ethernet-Kabel  
  Birne 3.LED  
  4,1 KΩ Widerstand  
  5.Female zu männlichen Überbrückungsdrähten  
  6.Bretbrett  
  7.Power Versorgung für Raspberry Pi  
  8. SD-Karte (8 GB oder höher)
* **Software-Implementierung**  
    
  1.Android Client App - spricht mit Firebase, aktualisiert Wert in Firebase.

2.Android Things App - liest Wert von Firebase und sendet Anweisungen an die Led-Lampe (durch Raspberry Pi-Gerät)

**Workflow :**

****

* Jedes internetfähige Android-Handy mit Android-App verbindet sich über die OAuth-Authentifizierung mit Firebase.
* Nach erfolgreicher Authentifizierung wird der Wert von Firebase aktualisiert / gelesen.
* Firebase aktualisiert die Werte auf IoT-Geräten mit AndroidThings OS und Android IoT-App.
* IoT-Gerät steuert alle angeschlossenen elektronischen Geräte.

* **IoT-Gerät in Cloud verbinden.**

1. Laden Sie die Developer-Vorschau von der Android Things-Website herunter.  
https://developer.android.com/things/preview/download.html  
  
2. Laden Sie Android Thing OS für Raspberry Pi herunter  
  
3. Nach dem Formatieren Ihrer SD-Karte müssen wir das Betriebssystem installieren.  
  
4. Nach dem Hochfahren stellt RaspberryPi eine Verbindung zu Ihrem Netzwerk über Ethernet her.  
  
5. Sobald es erfolgreich verbunden ist, sehen Sie die folgende Nachricht  
verbunden mit <deviceip-adresse>: 5555

6. LED positive Pin zu 7 (BCM4) und Negativ zu 9 (Ground)  
  
7. Erstellen Sie ein Firebase-Projekt unter https://firebase.google.com/.

8. Gehen Sie zum Abschnitt Regeln und ändern Sie die Regeln wie folgt

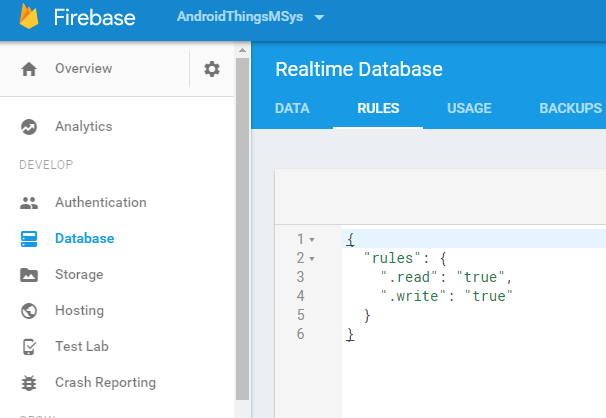
"rules": {

".read": "true",

".write": "true"

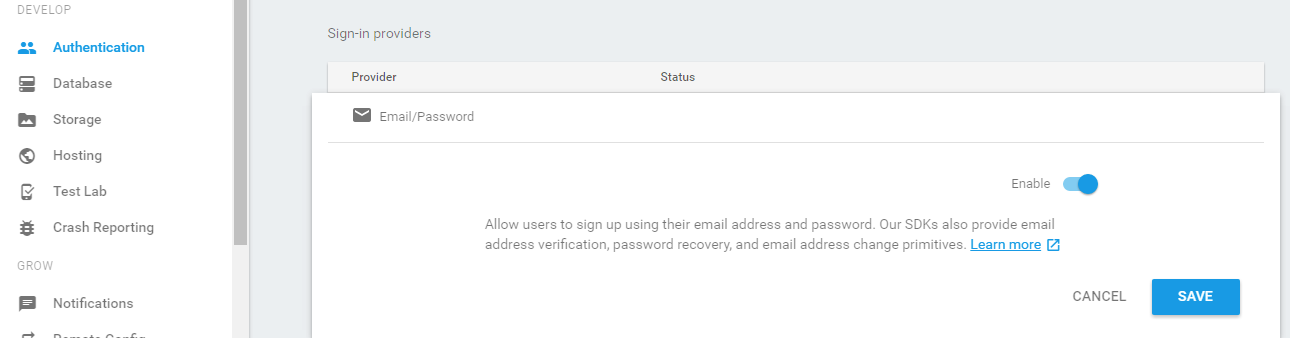
}

}



* Führen Sie nach der erfolgreichen Kompilierung Ihr erstes Android Things-Projekt aus, das mit Firebase konfiguriert ist.
* Klicken Sie im Android Studio auf die Schaltfläche "Ausführen" und wählen Sie Ihr Gerät aus.
* Nun wird Ihre Anwendung auf Ihrem Gerät laufen und Sie werden sehen, dass die Glühbirne blinkt.

Aktivirien E-Mail, Google, Facebook Login im Bereich Authentifizierung.



Fazit  
  
Das Internet der Dinge in Cloud Umgebung ist einfacher zu implementieren als der durchschnittliche Mensch denken würde. Wir haben großes Potenzial im IoT. Dies war nur eine einfache Projekt für IoT. Von Sicherheitssystemen bis zum Gesundheitswesen; Von Transportunternehmen bis zur Lagerverwaltung sind die Möglichkeiten in IoT einfach endlos. Mit dem Aufkommen von mobilen Apps wie IFTTT hat die Kommunikation die nächste Stufe erreicht. Wir können getrost sagen, dass die Zukunft dem IoT und Cloud Computing gehört.

**Verweise:**

[**https://developer.android.com/things/index.html**](https://developer.android.com/things/index.html)[**https://developer.android.com/things/hardware/raspberrypi.html**](https://developer.android.com/things/hardware/raspberrypi.html)[**http://fritzing.org/ (Electrical layout diagram tool)**](http://fritzing.org/)