Auteur: Max Huiskes (2151960)

Klas: 32022INF1

Vak Beroepsproduct 5/6

School: Avans Stratenlaan te Den Bosch

Datum: 24-10-2023

Versie: 1.0

Implementatie plan Data

Inhoudsopgave

[Inleiding 2](#_Toc149128126)

[Implementatie 2](#_Toc149128127)

[Data Ingestion 2](#_Toc149128128)

[Arduino 2](#_Toc149128129)

[TNN 3](#_Toc149128130)

[FiWare 3](#_Toc149128131)

[Data transformation 3](#_Toc149128132)

[Node Red 3](#_Toc149128133)

[Data storage 4](#_Toc149128134)

[MongoDB 4](#_Toc149128135)

[WebApp 4](#_Toc149128136)

[Samenvatting 4](#_Toc149128137)

[Bronnen 4](#_Toc149128138)

# Inleiding

In dit artikel schets ik de implementatiestrategie voor de datapijplijn. Hier zullen we de implementatiestrategie schetsen voor het opslaan, verwerken en opslaan van gegevens die zijn verzameld door LoRa-verbonden sensoren. Om de inname te implementeren, gebruiken we de Arduino, die is verbonden met de sensoren. De LoRa-gateway, die de gegevens opslaat, en het Things Network, dat de gegevens ontvangt. De transformatie vindt plaats met Node Red, dat de gegevens verwerkt met behulp van formules en algoritmen voordat deze worden opgeslagen in een MongoDB-database. De gegevens worden opgeslagen in een MongoDB-database en kunnen met behulp van een API uit een webapplicatie worden opgehaald.

# Implementatie

A diagram of a network

Description automatically generated

Zie bijlage B voor het bestand van de afbeelding

# Data Ingestion

## Arduino

Gegevens worden verzameld met behulp van een Arduino Uno die is verbonden met vier sensoren. Deze sensoren omvatten een temperatuursensor, een gewichtssensor, een radarsensor en een IR-breeksensor. De sensoren meten hun omgeving en sturen de gegevens naar de Arduino Uno, die de gegevens omzet in een payload-formaat met behulp van een LoRa-schild. Het schild slaat de gegevens op in een payload en verzendt deze. Om de sensoren te programmeren, moet u eerst de Arduino IDE op uw computer installeren en vervolgens de volgende plug-ins in de bibliotheekbeheerder installeren: Miles Burton's Dallas Temperature, Rob Tillaart's HX711, IBM's LMIC-framework en Jim Studt's OneWire.

A close-up of a circuit board

Description automatically generated

Zie bijlage A voor het bestand van de afbeelding

In de foto hierboven zie je cirkels met nummers er naast deze geven een onderdelen.

1: De gewicht sensor: Hx711 Gewichtssensor

2: De IR breakere sensor

3: LoRa-shield

4: Temperatuur sensor: Een DS18B20 Temperatuur sensor

5: Radar sensor: RCWL-0516 Radar sensor

De Arduino en zijn sensoren worden door de hele groep geïmplementeerd. Iedereen in de groep heeft zijn eigen sensor en heeft daar de verbinding en code voor gemaakt.

## TNN

Het Things Network ontvangt gegevens van de Arduino via een LoRa Gateway. U moet een account maken en een hub hebben om deel te nemen aan het Things Network. We hebben een payload formatter nodig om de data te gebruiken en leesbaar te maken.

De verbinding met het Things Network en de overdracht van sensorgegevens worden geïmplementeerd door de data-ingenieurs zodra de groep de sensoren op een Arduino heeft aangesloten.

## FiWare

FIWARE is het datawarehouse dat de gemeente 's-Hertogenbosch wil inzetten. We zullen uiteindelijk FIWARE gebruiken om een rest API te bieden om machine learning aan de applicatie toe te voegen. FIWARE is geïnstalleerd door de gemeente en de sensorgegevens worden nu geüpload.

De FIWARE-omgeving is eigendom van de gemeente 's-Hertogenbosch, hoewel de API is gemaakt door data engineers.

# Data transformation

## Node Red

Node Red is een programmeeromgeving die is ontworpen voor de ontwikkeling van IoT-toepassingen, die we zullen gebruiken om gegevens te converteren en op te slaan in een MongoDB-database.

We moeten eerst Putty installeren voordat we Node Red kunnen gebruiken. Na de installatie moeten we onze databasesleutel importeren door naar connection/SSH/Auth/Credentials te gaan. We komen hier om onze sleutel te kiezen en te importeren, en dan gaan we naar sessie en voeren ons IP-adres in een poort in.

Als u Node Red wilt gebruiken, moet u eerst een Oracle Cloud-account maken. Selecteer VM-instantie maken op uw Cloud dashboard en geef het een naam. Upload de SSH-sleutel die we ook met Putty hebben gebruikt. Selecteer het toewijzen van openbare IP-adressen, die we in Putty gebruiken om verbinding te maken met de database. We moeten werken aan de veiligheidslijst; om dit te doen, navigeert u naar subnet/beveiligingslijst en voegt u een ingress-regel toe voor poort 1880, die Node-RED vereist.

De data-engineers installeren Putty en genereren een sleutel om verbinding te maken met de Node Red-omgeving terwijl ze ook de virtuele machine maken.

# Data storage

## MongoDB

MongoDB is het platform waarop we onze database gaan bouwen. Om MongoDB te gebruiken, moet u eerst een account aanmaken op de MongoDB-website. Als u gegevens wilt bekijken en query's wilt uitvoeren op MongoDB, moet u eerst MongoDB-kompas downloaden en vervolgens de verbindingsreeks kopiëren die u van MongoDB hebt ontvangen. MongoDB Atlas moet worden gebruikt om de MongoDB-database buiten in plaats van intern te hosten. Maak een account voor Atlas aan. Maak een cluster aan om te gebruiken als ontwikkelomgeving. Voer uw huidige IP-adres in Atlas in om toegang te krijgen tot uw database. Maak een databasegebruiker aan en maak verbinding met uw database via Node.js of Compass.

De MongoDB-database en het MongoDB-kompas worden geïnstalleerd en geconfigureerd door data-engineers met behulp van sensorgegevens. De databasebeveiliging wordt geïmplementeerd door een beveiligingsspecialist.

## WebApp

De WebApp is de applicatie die sensordata uit de database opslaat en beschikbaar stelt. We hebben Vue.js en een Azure-server nodig om aan de slag te gaan. We gebruiken Azure om de web toepassing te hosten. Hiervoor maken we eerst een Azure-account op de Azure-website. We moeten de Azure-site ook gebruiken om een resource met een unieke naam te maken en het besturingssysteem op te geven waaraan het moet werken. We hebben GitHub nodig om de web-toepassing op Azure te bouwen en we moeten de omgeving een domeinnaam en DNS-configuratie bieden.

De front-end expert implementeert het maken van de webApp en de Azure omgeving, en de front-end zorgt ervoor dat de sensordata duidelijk getoond wordt. De beveiligingsspecialist zorgt ervoor dat de verbindingen veilig zijn. De data engineers maken de koppeling tussen de front-end en de database.

# Samenvatting

In dit artikel wordt ingegaan op de implementatie van de datapijplijn. Het beschrijft de vele onderdelen van de pipeline en wat er nodig is om deze te implementeren. De Arduino Uno wordt gebruikt om sensorgegevens te verzenden. Het Things Network ontvangt gegevens en stuurt deze naar de door de gemeente opgegeven FIWARE-omgeving en Node-Red. Node-Red transformeert de gegevens en stuurt deze naar de MongoDB-database. Om Node-Red te gebruiken, moeten we de SSH-client Putty gebruiken en onze sleutel en het IP-adres opgeven. De sensorgegevens worden opgeslagen in een MongoDB-database en zijn toegankelijk na installatie van het Compass-programma. We gebruiken Atlas om de externe database te verbinden met een cluster en ons IP-adres te verstrekken. De web toepassing is gebouwd in Vue.js en gehost op een Azure-server. Azure host de webapp met behulp van een Azure-interface die de webapp importeert vanuit GitHub.

# Bronnen

*Get Started with Atlas — MongoDB Atlas*. (n.d.). <https://www.mongodb.com/docs/atlas/getting-started/>

Sharp, T. (n.d.). *Installing Node-RED In An Always Free VM On Oracle Cloud*. <https://blogs.oracle.com/developers/post/installing-node-red-in-an-always-free-vm-on-oracle-cloud>

# Bijlage A – Arduino uno foto



# Bijlage B – Datapipeline

