# Hva bruker jeg?

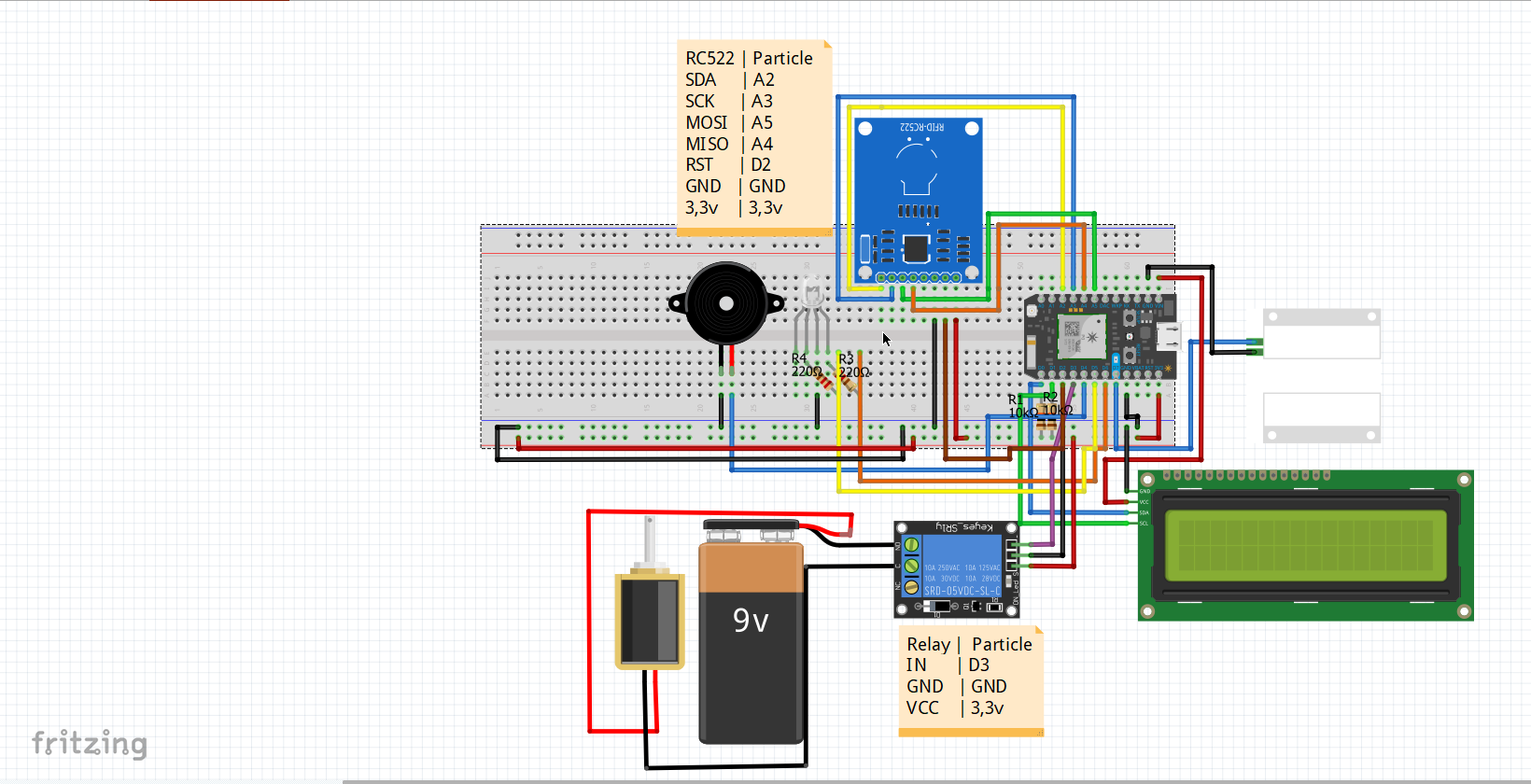
Her har jeg valgt å bruke particle photon som microkontroller, siden den har wifi og er enkelt å sette opp mot nettet. Samt kan den integreres mot andre støttende tjenester som microsoft azure eller google cloud. Jeg tok i bruk google cloud sine tjenester og bruker google pub/sub for å sende data fra particle photon til google cloud datastore, hvor data blir lagret i en database. Google cloud pub/sub er bra fordi den kan raskt hente og sende data til forskjellige enheter asynkron du kan lese mer om dette hos google cloud sine dokumentasjoner. Jeg velger også å bruke Nodejs express som server for å snakke til google cloud sin api. Som front-end bruker jeg reactjs for å vise fram dataene jeg lagrer fra google datastore.

**Embedded:**  
Jeg bruker en magnetisk dør sensor med en buzzer. Når en åpner døren for lenge, eller prøver å åpne døren uten få godkjenselse fra rfid leseren. Vil buzzer lage lyder, slik at noen kan komme og stoppe personen. RFID leseren skal lese chip/kort og sjekker om det er riktig kort eller ikke for å komme inn.   
Solenoid lock-style er for å låse døren eller skapet. Bruker også en strøm reelle for å gi solenoid lock det den trenger av strøm for å åpnes. LCD skjermen skal vise tid, dato og skal gi beskjed om man har tilgang eller ikke, etter at man har skannet rfid kortet sitt. Vis rfid leseren gir kortet ditt godkjent så skal rgb-led lyse grønt, mens ikke godkjent lyser rødt.

# Hvordan setter jeg ting opp?

**Komponenter:**  
1x Buzzer,  
1x 9v Battery,  
1x RGB led,  
1x 9V battery,  
1x RFID-RC522,  
1x Solenoid lock-style,  
1x Magnetic door switch,  
1x srd-05vdc-sl-c relay,  
1x LCD 16x2 QAPASS,  
2x 10k ohm resistor,  
2x 220ohm resistor.  
2x RFID kort/chip

**Koblingskjema:**



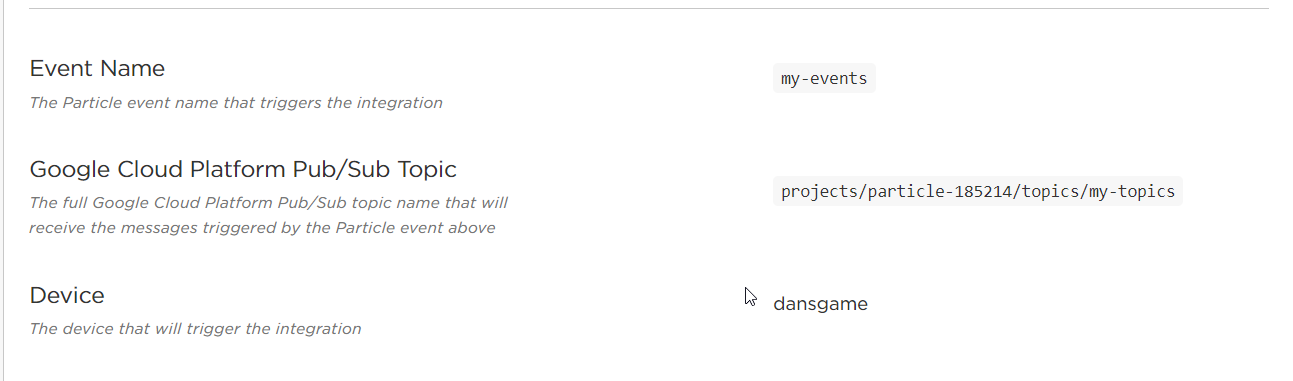
**Pins:**  
Magnetic door sensor – D7,  
Buzzer - D4,  
RGB LED – D5, D6.

**srd-05vdc-sl-c relay:**Den som følget med kittet er på kinesisk.   
Slik skal den kobles.  
  
**NO – 常开 – 9v battery.  
COM – 公共端**  **– solenoid lock-style.**  
**NC – 常关 – Nothing.**

**LCD:**Trenger 10k motstand på pin D0 og D1 fra 3.3v

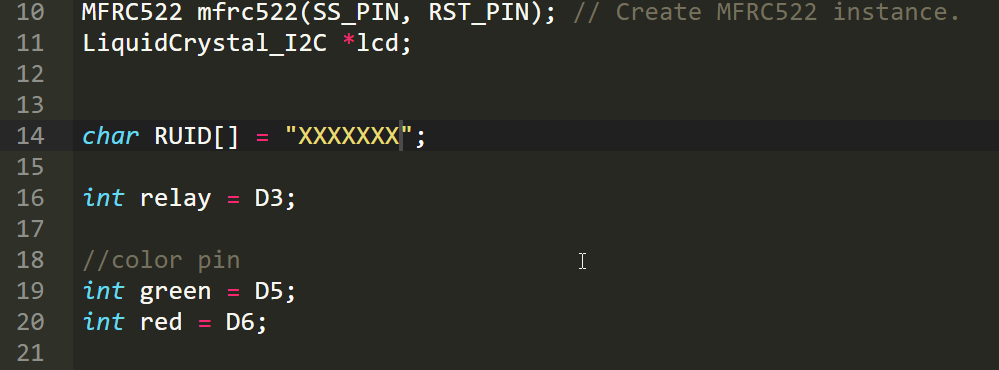
**Anbefalt programversjon:  
NodeJS: v6.11.3**<https://nodejs.org/download/release/v6.11.3/> **integrer particle photon med google cloud platform:  
Følg hele denne guiden for teste pub/sub og lagre data på datastore.**<https://docs.particle.io/tutorials/integrations/google-cloud-platform/><https://docs.particle.io/tutorials/integrations/google-cloud-platform/#storing-data-in-a-datastore-database>

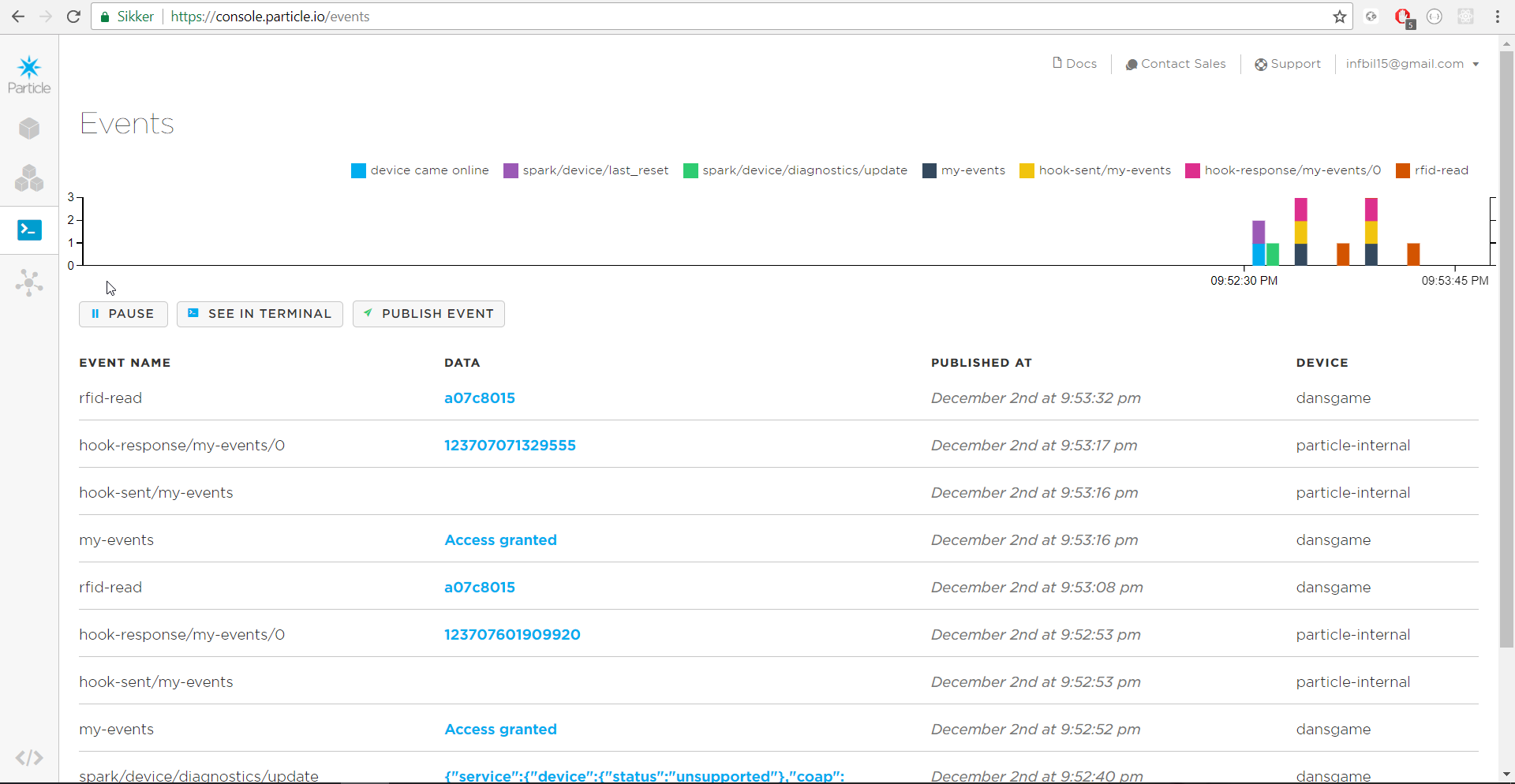
**Bruker event navnet «my-events» når jeg skal gi data videre til pub/sub**

****

****

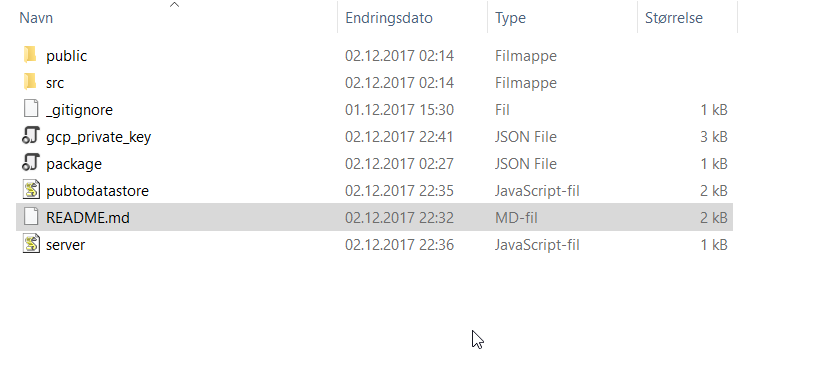
**Particle code:  
Bruker Particle sin web ide til å kode.  
navnet på kode filen heter kode.ino så kopier koden inn til particle ide.  
RUID skal inneholde data på et av rfid kort/chippene du vil at skal ha tilgang til dører ect.**

****

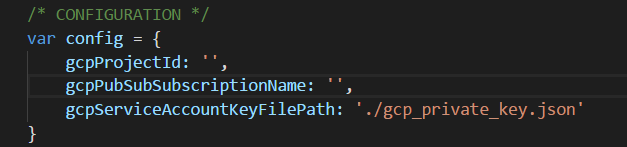
**For å finne RFID data så trenger du bare å gå på particle sin console og skanne inn et av kort/chip. Data vil se slik ut. **

**Kopier deretter data fra rfid-read til RUID[] = «xxxxxxxx»**

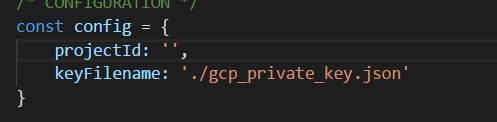
**Hvordan får jeg opp reactjs, node express, pubsub?  
Dette krever at du har laget en private key fil fra tidligere instrukser og testet pubsub to datastore:**<https://docs.particle.io/tutorials/integrations/google-cloud-platform/#storing-data-in-a-datastore-database><https://github.com/spark/google-cloud-datastore-tutorial>

****

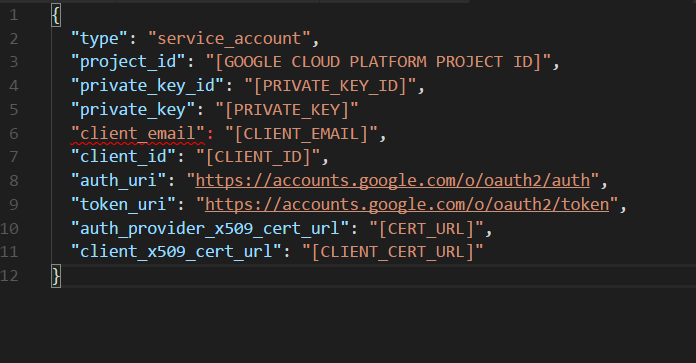
**Fyll informasjon i pubtodatastore.js filen.  
du finner prosjektid finner du i google cloud sin dashboard og er navnet på ditt prosjekt og id nummer.   
Subscription navn finner du i pub/sub avdelingen og er navnet på der du ønsker å subscriber på for henting av data fra pubsub.**

****

**Fyll informasjon i server.js filen  
finner prosjektid samme som tidligere i google cloud dashboard og er id navnet i prosjektet ditt.**

****

**Endre gcp\_private\_key.json fra din egen private key fil.**

****

**Kjøring av react-app folder steps:  
1. åpne terminal i samme folder som prosjektet ligger.   
2. npm install  
3. node pubtodatastore.js  
4. node server.js  
5. npm start**

**Har laget en video på hvordan man kjører dem, men husk å npm install før du kjører disse filene.  
  
React app:  
Bruker react app for å vise fram data historikk og når noen har nettopp skannet kortet.  
Den henter data hvert 5 og 10 sekund fra google cloud datastore. Så man må passe på sin daglige kvote hos google-cloud. Om man registrerer for gratis trial hos google-cloud så får man 300$ credits som kan hjelpe vis man har fullt opp noen av kvotene.**

**Server.js:   
Henter data fra datastore.  
  
pubtodatastore.js:  
Henter data fra particle photon og setter data i en topic. Datastore lagrer topics data i sin datastore.**

# **Veien videre**

Videre ønsker jeg å implementere mer funksjonaliteter til front-end og backend. hvor jeg har en logging system og kan endre tid på alarm systemet. Vil også vise fram dataene i en graf over tid. For å få en bedre oversikt over tidligere data. Annet jeg ønsker å utvikle er en egen mobil app laget i react native. Den appen ha en logging system slik at man kan åpne dører eller skaper fra mobilen i stedet for rfid. Den skal kunne også bruke pub/sub til google-cloud for å hente data som skjer der og da. Vil også implementere en notifikasjon system på når uautoriserte kort blir lest opp av rfid leseren.

**kilder:  
pubsub:**<https://cloud.google.com/pubsub/docs/quickstart-client-libraries#pubsub-client-libraries-nodejs> **datastore:**<https://cloud.google.com/datastore/docs/reference/libraries> **particle photon:**<https://docs.particle.io/tutorials/integrations/google-cloud-platform/> **store pubsub to datastore:**<https://github.com/spark/google-cloud-datastore-tutorial>