

Huidige situatie

- Een beveiligde webservice(NoSi) die een verbinding heeft met een MongoDB database.
- NoSi ondersteunt verschillende get en post requesten om data te manipuleren, zo ook om een historie van data uit te lezen.
- Een kleine Angular 5 front-end applicatie die objecten data toont uit NoSi
- NoSi verpakt in een Docker container
- Een homeLynk systeem met een eigen database die data uit het huis opslaat in deze database (Schneider Electric)
- Een tweede webservice (FlaSi) die binnen het Selficient huis draait om het homeLynk systeem aan te sturen.
- FlaSi ondersteunt de mogelijkheid om rauwe MySQL query's uit te voeren en deze data terug te krijgen

Ideale situatie

Wanneer het systeem af is moeten de volgende componenten de volgende onderdelen kunnen ondersteunen.

FlaSi

- Er is een endpoint beschikbaar die meegegeven MySQL query's uit kan voeren en het resultaat van deze terugstuurt naar de aanvrager
- Aan de hand van adressen kan FlaSi het HomeLynk systeem aansturen

NoSi

- De data uit de query's van FlaSi wordt opgeslagen op een eigen server die in eigen beheer is. Deze data wordt dusdanig opgeslagen dat het verkeer tussen de front-end applicatie en FlaSi zo minimaal mogelijk is.

Front-end services

- De Angular 5 front-end applicatie is beveiligd op gebruikersnaam en wachtwoord niveau
- De Angular 5 front-end applicatie toont de opgeslagen data van interacties via grafieken
- De Angular 5 front-end applicatie toont verschillende voorbedachte grafieken met data uit één of meerdere huizen

Opdracht inleiding

Het Selficient huis geniet na een succesvolle start veel aandacht. Voor dit project komen er nieuwe mogelijkheden aan om het huis te laten zien aan de wereld en aan wedstrijden mee te doen. Zo komt er een volgende editie van de Solar Decathlon challenge aan in Hongarije, maar ook in Californië zijn ze geïnteresseerd in wat deze studenten gebouwd hebben. Ook de gemeente Utrecht wil graag helpen en heeft plek voor het huis vrijgemaakt op de Uithof, waar deze het komende jaar ook opgebouwd gaat worden.

Zoals je ziet moet het huis op veel verschillende plekken zijn, en dit is nog zonder te talloze congressen en promotiepraatjes. Voor dit probleem heeft een vorig ICT-team bedacht om het huis te gaan virtualiseren, maar dan wel met een speciaal vleugje creativiteit.

Dankzij de Oculus Rift kun je met een controller door het huis lopen om zo precies het huis vanbinnen mee te kunnen maken. Je loopt door deuren, zet lampen aan en kijkt nog eens goed naar de modulaire muren van het Selficient huis. Het verschil zit hem er alleen in dat wanneer jij je lamp aan doet in de virtuele wereld, in het echte huis ook de lamp aanspringt. *Deze twee werelden zijn verbonden!*

Het vorige team heeft voor ons een mooie basis gelegd en heeft alles zo bewerkbaar mogelijk gelaten. Er is veel documentatie voor jou en je team beschikbaar en daarnaast ook ondersteuning van het vorige team waar nodig.

Opdracht omschrijving

Op het moment van schrijven worden alle handelingen die een connectie hebben met de echte wereld opgeslagen door middel van een REST-api naar een MongoDB database. Dit kan voor het Selficient huis interessante data zijn om te kunnen analyseren. Op het moment is deze functionaliteit om te analyseren er nog niet. Er zijn wel endpoints beschikbaar om de data op te vragen, maar deze moet nog op een manier getoond worden aan de eindgebruikers. Op dit moment is er een klein dashboard beschikbaar, geschreven in Angular 5, die een connectie heeft met de API. Deze dient dus met meer analytische mogelijkheden te worden uitgebreid.

Deze extra functionaliteiten dienen ook query's te zijn die gebruik maken van de FlaSi API die de data voor jullie kan ophalen uit het Selficient huis. Om dit effectief te doen zal deze data op een bepaalde manier vanuit de MySQL database moeten worden getransporteerd naar een MongoDB database, zodat je niet constant verkeer hebt tussen het huis en de front-end applicatie.

Daarnaast is dit dashboard nog niet beveiligd; ook hier een uitdaging voor jullie. Zorg dat alleen geregistreerde gebruikers bij deze informatie kunnen. En zorg dat deze registratie functionaliteit wordt verwerkt in de huidige MongoDB omgeving, we willen immers alles zo centraal mogelijk houden.

Applicatie data flow

Hieronder is een kleine schets van hoe de data loopt tussen de verschillende delen van de applicatie. Deze opdracht haar focus ligt voornamelijk op de backend service en het Selficient Huis haar node.

