

S(N)E - FYS DEELTIJD
FYS DESIGN DOCUMENTATION

Hogeschool van Amsterdam
HBO-ICT Deeltijd - Software Engineering

Versie: 0.1
Datum: 12-06-2018

Auteurs: Milo Gisolf
Nick Vlug
Daniël Kraaijeveld
Wesley Langelaan
Robert-Jan Buddenböhmer
Abdel El-Medny

Studentnummers: 500690961
500775497
500783668
500779152
500790846
500789984

Docent: J. Somers
Vak: S(N)E - FYS Deeltijd

Inhoudsopgave

Inhoudsopgave	2
Colofon	3
Media formats	4

Colofon

Document: **S(N)E - FYS DEELTIJD**
FYS DESIGN DOCUMENTATION

Docenten: J. Somers
Docent technisch FYS
e. j.somers@hva.nl

Studenten: Milo Gisolf - 500690961
Student HBO-ICT SE Deeltijd
e. milo.gisolf@hva.nl

Nick Vlug - 500775497
Student HBO-ICT SE Deeltijd
e. nick.vlug@hva.nl

Daniël Kraaijeveld - 500783668
Student HBO-ICT SNE Deeltijd
e. daniel.kraaijeveld@hva.nl

Wesley Langelaan - 500779152
Student HBO-ICT SNE Deeltijd
e. wesley.langelaan@hva.nl

Robert-Jan Buddenböhmer - 500790846
Student HBO-ICT SNE Deeltijd
e. robert-Jan.buddenbohmer@hva.nl

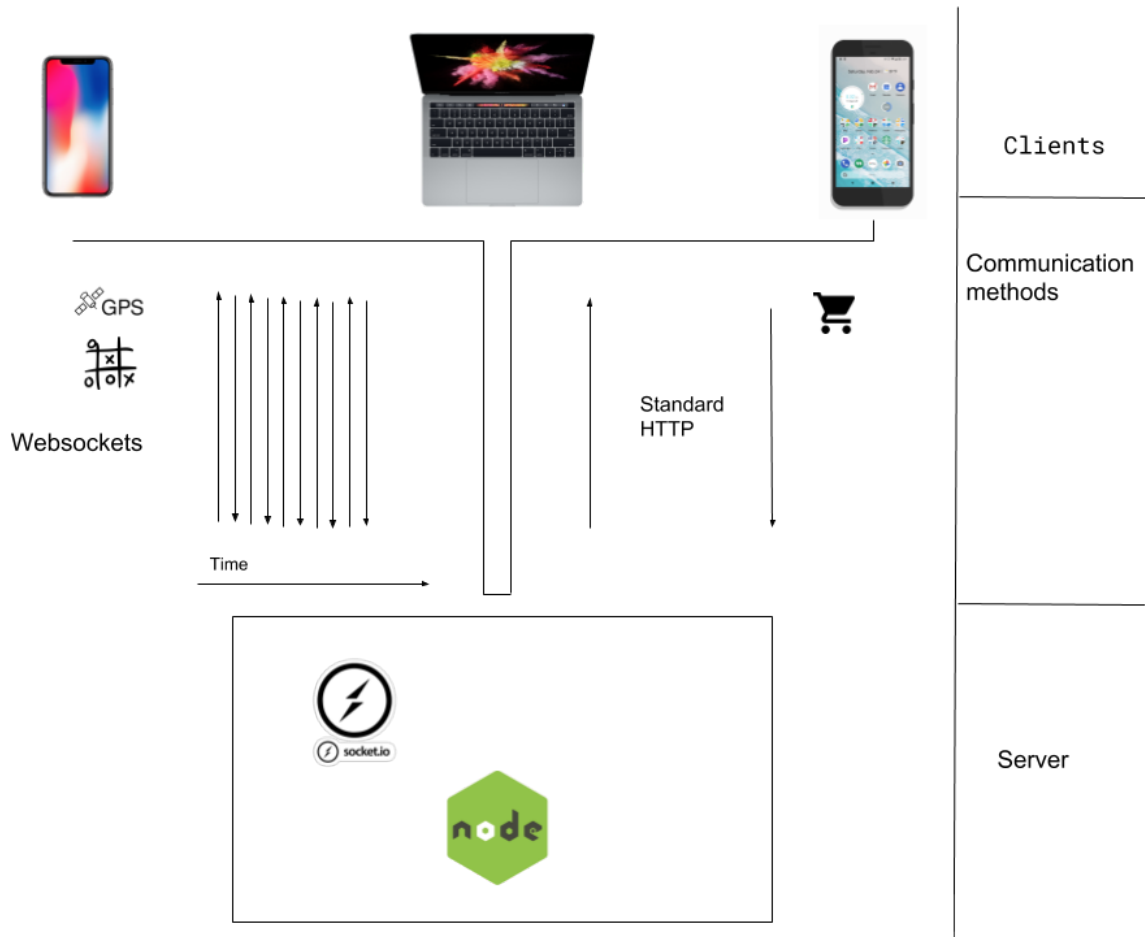
Abdel El-Medny - 500789984
Student HBO-ICT SE Deeltijd
e. abdel.el-medny@hva.nl

Media formats

Onze server voor de Captive Portal draait op Apache 2 en PHP 7.1. PHP verwerkt aanvragen en toont vervolgens door middel van Apache2 HTML opmaak. PHP verwerkt verschillende aanvragen voor nu de login flow / registratie flow.

Onze server voor de FYS omgeving draait door middel van express in Node JS. Express is modulair opgebouwd met verschillende modules. Het ondersteunt de standaard web extensies en json body parser. De front-end is gebouwd met Angular en bootstrap, dit is uiteindelijk HTML, typescript en CSS en hier en daar wat media bestanden. Zo kan er zonder enige moeite uit de static content route video- en audiomateriaal gehaald worden.

Daarnaast werkt onze HTTP listener ook met een Socket IO hook, dit houdt in dat wij met de web sockets protocol data kunnen versturen en ontvangen. Zo kunnen wij GPS informatie overdragen i.p.v. een oude techniek ook wel pulling via Ajax. Socket IO heeft als voordeel dat wij informatie naar alle peers kunnen versturen, zonder dat ze het opvragen. Met HTTP zouden onze clients continue de backend moeten pollen voor updates.



Client - server flow

Door deze simpele opzet, kunnen browser ook onderling communiceren echter is dat wel via de server, aangezien browser zelf geen P2P verbinding ondersteunen.

Wij hebben Socket IO op het moment toegepast voor twee belangrijke features in de FYS applicatie:

1. Game boter kaas en eieren
2. GPS overzicht.

Omdat we socket.io gebruiken, was het belangrijk voor ons om reactief programmeren technieken toe te kunnen passen. Dit hebben we kunnen doen op de front end en de backend door middel van rxjs te gebruiken.