TELE-HEALTH

**Table of Contents**

[Inleiding 1](#_Toc431847304)

[Aantal personen: 3 1](#_Toc431847305)

[Vereisten 1](#_Toc431847306)

[Front End 1](#_Toc431847307)

[Back End 2](#_Toc431847308)

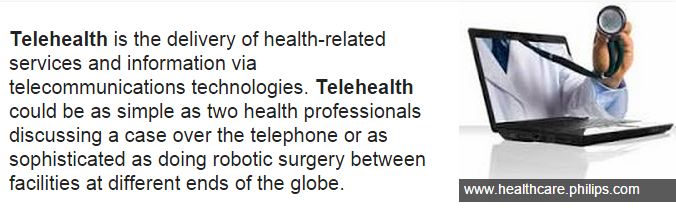
[Uitwerking 2](#_Toc431847309)

[Te gebruiken technologieën 2](#_Toc431847310)

[Vereisten 2](#_Toc431847311)

# Inleiding

De term “telehealth” is zeker niet meer nieuw:



Telehealth legt bovendien meer nadruk op monitoren en preventieve gezondheidszorg eerder dan op revaliderende geneeskunde na een ziekte. De implementatie ervan gebeurt eerder langzaam. Daarom gaan we zelf een case programmeren. We bestuderen en kopen een aantal minder dure IoT devices, die toelaten een gezondheidsaspect te monitoren. Bijvoorbeeld: hartslag, bloeddruk, slaapstoornis of gewoon temperatuur. Je kiest maar.  
Vandaag worden die zaken vaak opgeslagen in een offline device, waarbij de patiënt verschillende doktersbezoeken aflegt. We vervangen dit door real time devices die continu de informatie doorsturen naar de betreffende geneesheer. De communicatie tussen device en mobile telefoon verloopt bvb. via bluetooth. De telefoon doet de rest (WiFi). De geneesheer kan rapporten opvragen of krijgt warnings.

Noot voor een mogelijke testomgeving. Wat gebeurt er met de hartslag van studenten tijdens een examen?

## Aantal personen: 3

# Vereisten

## Front End

De front van de applicatie bestaat uit verschillende delen.

* De welkom pagina toont een overzicht van een reeks eerder ingestelde devices,die de geneesheer specifiek monitort in real-time.
* Naast de real-time weergave worden op vooraf ingestelde intervallen de gemonitorde waarden van een patiënt opgeslagen en weergeven
* Warning pagina met een hoorbaar geluid, wanneer een nieuwe warning binnenkomt.
* Filter pagina met verschillende combineerbare criteria: device type, patiënten, regio, periode.
* Bewaren van een rapport in pdf.

## Back End

* Alle data wordt opgeslagen en is voorzien van een timestamp.
* Je kan in real-time een persoon selecteren om er mee te communiceren.
* De geneesheer kan gegevens back-uppen, zodat ze tijdelijk niet meer zichtbaar zijn in het overzicht.
* De geneesheer kan een patiënt toegang geven tot zijn persoonlijk patiënt rapport.

# Uitwerking

## Te gebruiken technologieën

|  |  |
| --- | --- |
| **Front End** 8pt van de 20pt | **Back End**\*7pt van de 20pt |
| * CSS Preprocessor, * taskRunnner, * front -end testing, * LocalStorage, * Maps, * gebruik van framework naar keuze (angular). | * volledig asynchroon in javascript – node.js, * persistentie in een NoSQL database (mongodb), * sockets voor multi-user integratie(socket-io), * autorisatie, authenticatie, * minstens één integratie en één unittest(jade), * deployment op cloud naar keuze(azure, heroku), * gebruik van framework naar keuze (express). * IoT device   \* Default zaken, zoals gezien tijdens de lessen, staan tussen haakjes maar mogen aangepast worden. |

Noot:

* Wie geen Front End volgt kan gebruik maken van een standaard HTML, Bootstrap of Express lay-out.   
  Eventueel kan een student, die enkel Front End volgt, deze front-end aanmaken als project.
* Wie geen Back End volgt kan alle data stockeren in een lokale JSON file.  
  Eventueel kan een student, die enkel Back End volgt, deze backend aanmaken als project.

## Vereisten

* Programmeer in het Engels
* Javascript volgens de regels van de kunst (jshint als evaluatie tool)
* CSS volgens de regels van de kunst (csslint als evaluatie tool)
* Conventies:
  + Gebruik de juiste css naamgevingen.
  + Gebruik duidelijke code.
* Performantie:
  + minifying,
  + concat
  + performante selectoren,
  + performante javascript,
  + foutcontrole op kritische onderdelen,
  + zoveel mogelijk DRY en abstract (= vermijden van herhalingen)
* Mooie, acceptabele (UI) en gebruiksvriendelijk(UX) userinterface voor zowel mobile of desktop gebruik.
* Gedocumenteerd:
  + Waar nodig, als verduidelijking, in de code
  + Aanvullend document (min. 2 bladzijden) met de nodige paswoorden, online URL’s, werkverdeling, schatting van werkuren, grootste moeilijkheden(2) en successen(2), eventueel architectuur en database schema, eventuele referenties en persoonlijke conclusie
* Project beschikbaar op een source control systeem naar keuze.
* Indien datum: ten laatste afgewerkt in de week vóór het examen. Tussentijdse evaluaties zijn mogelijk.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_