

Functioneel/Technisch ontwerp



**well NETWORKED
top CONNECTED**

business models • smart services • strategies • solutions

Contents

Contents	1
Move4Vitality Platform	2
React Native App.....	2
Android.....	2
IOS	2
React Web Applicatie	2
REST Web API.....	2
Databases.....	3
NoSQL.....	3
SQL	3
C Level Diagram	4
C1 Diagram.....	4
C2 Diagram.....	4
C3 Diagram.....	5
C4 Diagram.....	6
Flowchart.....	7
User Stories	8
Moscow	9
Use Cases.....	10
Als deelnemer moet ik kunnen inloggen via Google.	10
Als deelnemer moet ik toestemming kunnen geven om mijn gegevens te delen.....	10
Als fysiotherapeut wil ik gezondheidsdata kunnen inzien via een dashboard.	11
Als fysiotherapeut wil ik meerdere grafiek opties bij het weergeven van klant data.....	11
Sequence Diagram.....	12
Als deelnemer moet ik kunnen inloggen via Google/ Als deelnemer moet ik toestemming kunnen geven om mijn gegevens te delen.	12
Sequence diagram voor data die verzameld en verwerkt word.(dit is het sequence diagram voor alle user stories die data verzamelen en verwerken.)	12
Smart Watch diagram directe koppelingen	13
Meest Optimale implementatie	14
Voor & Nadelen.....	14
Google Fit	14
Apple Health.....	14
Strava.....	15

Move4Vitality Platform

Bij het begin van het project heb ik besloten om alle technische onderdelen van het project te noteren zodat mij duidelijk word waar alles voor gebruikt word.

React Native App

De applicatie biedt de mogelijkheid voor de deelnemer om in te loggen met hun eigen account. Na inloggen kan de gebruiker de app koppelen aan Apple Health op iOS of aan Health Connect en/of Google Fit op Android. De app haalt gezondheids- en activiteitsgegevens op van deze koppelingen en stuurt deze vervolgens naar mijn API. Ik maak gebruik van Expo Go om de applicatie te testen. Echter, vanwege het ontbreken van ondersteuning voor native code in Expo Go, vereist het bouwen van de applicatie aparte procedures.

Android

Om deze applicatie op android te kunnen bouwen is een Windows omgeving of een Mac omgeving nodig hiervoor gebruik ik mijn persoonlijke laptop. Om deze applicatie te kunnen testen gebruik ik de android telefoon die ik verkregen heb van mijn stagebedrijf.

IOS

Om deze applicatie op IOS te kunnen bouwen is een Mac omgeving nodig hiervoor ben ik met Roy in gesprek gegaan over de opties en er is nog een Mac book beschikbaar van het bedrijf waarbij de accu vervangen moet worden die ik kan gebruiken zodra dit gereed is. Om deze applicatie te kunnen testen gebruik ik mijn eigen iphone.

React Web Applicatie

De webapplicatie is essentieel voor het tonen van activiteits- en gezondheidsgegevens aan fysiotherapeuten. Hier hebben fysiotherapeuten de mogelijkheid om deelnemers te selecteren en relevante gegevens op te vragen. Deze data wordt opgehaald vanuit mijn API.

REST Web API

De API van Move4Vitality is een C# API die werkt volgens Clean Architecture. Deze architectuur volgt het Dependency Inversion Principle en Domain Driven Design. Clean Architecture plaatst de business logica en het applicatiemodel in het midden van de applicatie. Infrastructuur- en implementatiedetails zijn afhankelijk van de Application Core, bereikt door het benoemen van abstracties en interfaces in de infrastructuurlaag. De structuur van de backend ziet er uit als volgend:

- API bevat de webgerelateerde onderdelen: controllers, web specifieke services, configuratie, enz.
- Applicatie: De applicatielaag bevat de bedrijfslaag hierin staat alle logica.
- Gemeenschappelijk: Bevat hulpprogrammaklassen en -services en is de enige assembly waarnaar verwezen kan worden door alle andere assemblies (en mag geen afhankelijkheid hebben van een andere projectassembly!).
- Domein: Bevat domeinentiteiten (EF POCO's).
- Infrastructuur: Implementaties voor externe afhankelijkheden (Auth0, Garmin, enz.).
- Persistentie: Bevat EF-configuratie en definitie van de DB-context."

deze API ontvangt activiteiten en gezondheids data vanuit mijn React Native App. Deze data word vervolgens verwerkt en opgeslagen in een database.

Databases

NoSQL

Als database voor dit project word gebruik gemaakt van een NoSQL database voor het opslaan van alle Garmin gezondheidsdata die Move4Vitality ontvangt. De reden hiervoor is dat er veel data opgeslagen kan worden, wat je gaat hebben als een activiteitentracker constant data doorstuurt. Voor mijn Proof of concept heb ik alleen een Cosmos database gebruikt omdat Ruben mij dit had aangeraden en hun hadden er nog een beschikbaar.

SQL

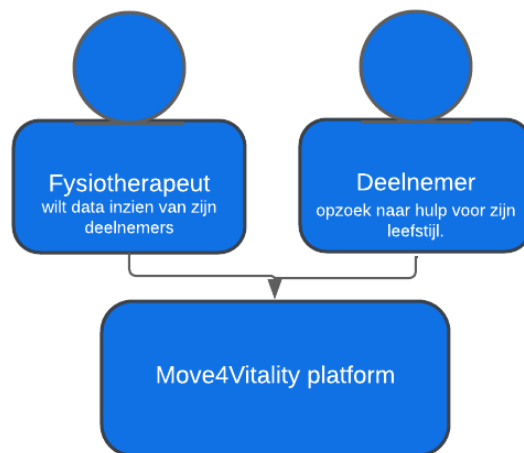
Ook word er gebruik gemaakt van een SQL database in het platform dit om het totaal aantal dagelijkse stappen en stappen per activiteit op te slaan. De gegevens in de SQL database zijn cruciaal voor de functionaliteit van het domein. Binnen het domein zijn bepaalde relaties van belang waardoor er gekozen is voor een SQL database.

C Level Diagram

Vervolgens is een C Level diagram gemaakt om een duidelijk beeld te krijgen van het gehele Move4Vitality systeem en alle externe programma's die hier mee te maken hebben.

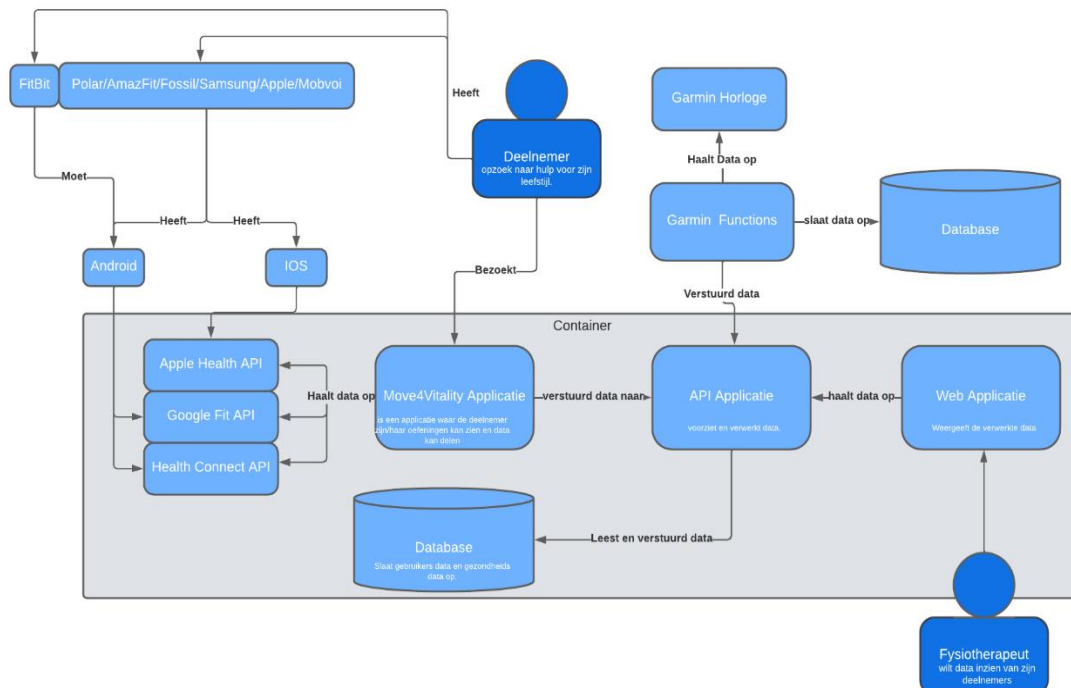
C1 Diagram

Dit diagram weergeeft welke gebruikers interactie hebben met het Move4Vitality Platform



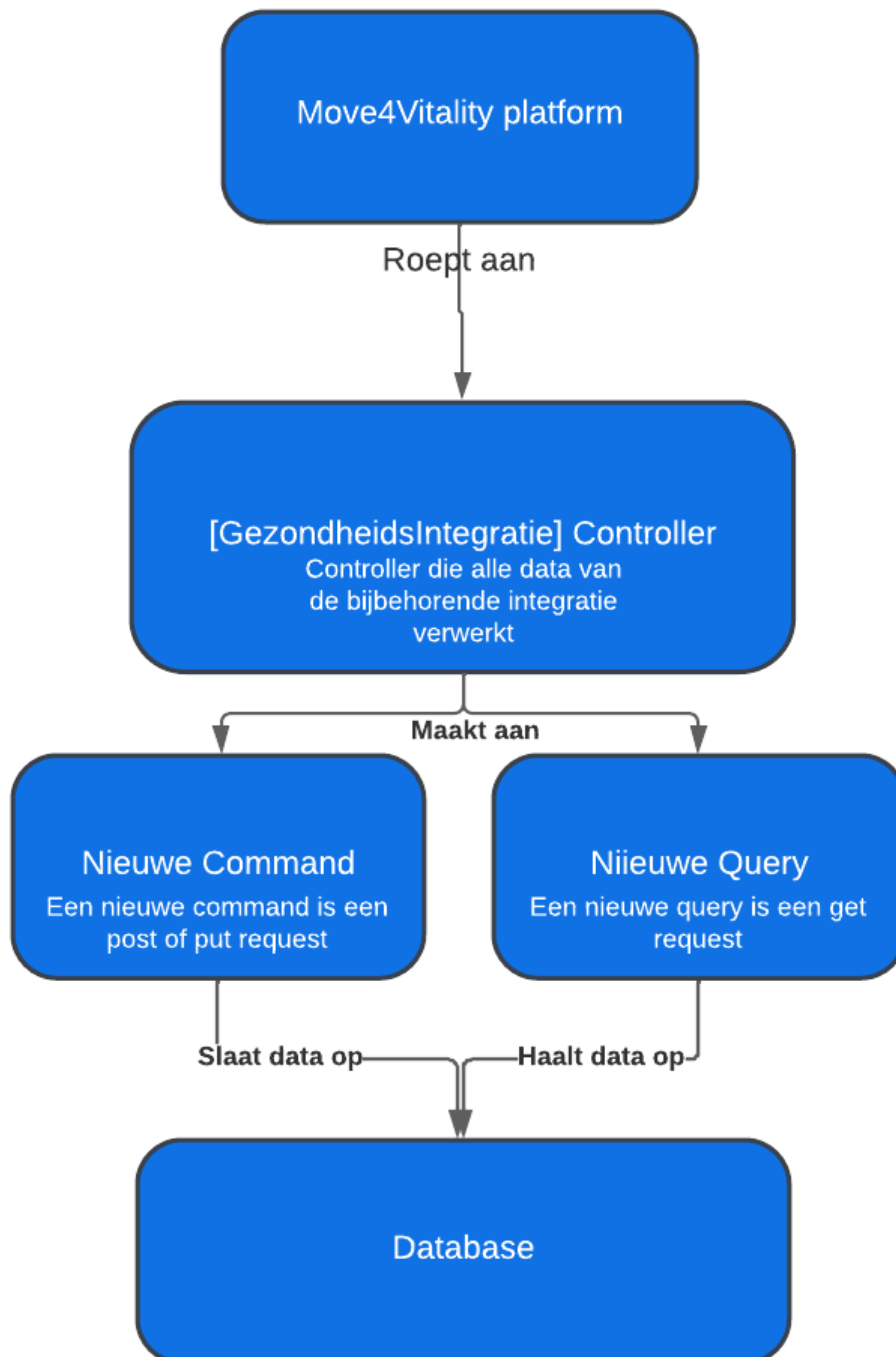
C2 Diagram

Dit diagram weergeeft het gehele Move4Vitality systeem hoe de personen, programmas/apparaten hiermee in aanraking komen.



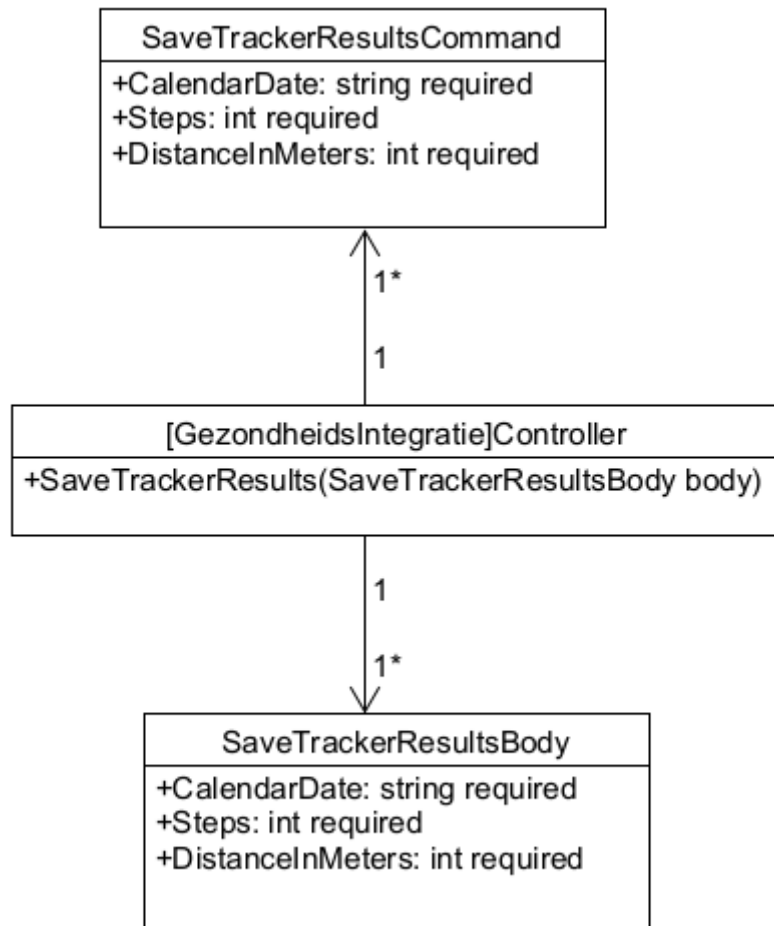
C3 Diagram

Het C3 diagram weergeeft hoe de API van Move4Vitality in elkaar zit, de applicatie of website roept een controller url aan deze maakt vervolgens een Command of Query aan een command is een post of put request en een Query is een get request deze voeren vervolgens een handler uit die de data ophaalt of opslaat in de database.



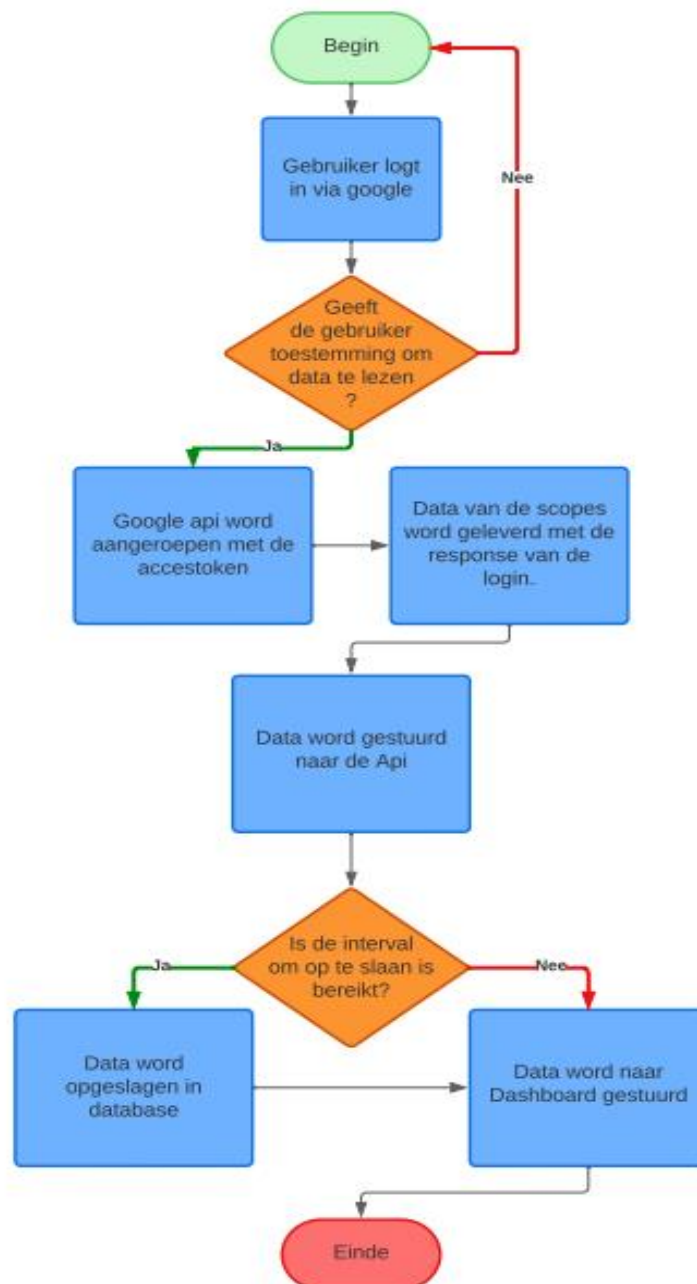
C4 Diagram

In dit diagram zie je welke classes worden gebruikt door de controller en wat de variabelen zijn in deze classes. De controller heeft een `SaveTrackerResultsMethod` die een `SaveTrackerResultBody` ontvangt wat een `CalendarDate`, `Steps` en `DistanceInMeters` bevat dit word vervolgens omgezet naar een `SaveTrackerResultsCommand` waarin de data word opgeslagen in de database.



Flowchart

Ik heb besloten om een flowchart te maken om het process in beeld te brengen hoe de connectie met google fit werkt. Zo kan ik Roy duidelijk laten zien hoe ik de applicatie wil laten werken.



User Stories

Hier is een lijst gemaakt van alle functionaliteiten die geïmplementeerd moeten/kunnen worden. In het huidige systeem van Move4Vitality, ook heb ik een complexiteit aan deze functionaliteiten toegevoegd zodat er een schatting is van hoelang ik met de functionaliteit bezig ben

Functionaliteit	Complexiteit
Als deelnemer moet ik mijn apple gezondheid kunnen koppelen aan de app op ios	12
Als deelnemer moet ik mijn google fit kunnen koppelen aan de app	12
Als deelnemer moet ik mijn health connect kunnen koppelen aan de app op android.	12
Als deelnemer moet ik toestemming kunnen geven om mijn gegevens te delen.	5
Als fysiotherapeut wil ik ik gezondheidsdata kunnen inzien via een dashboard.	13
Als fysiotherapeut wil ik dat de dagelijkse stappen verzameld en verwerkt worden.	5
Als fysiotherapeut wil ik dat de afstand afgelegd door een gebruiker verzameld en verwerkt word	5
Als fysiotherapeut wil ik meerdere grafiek opties bij het weergeven van klant data	13
Als fysiotherapeut wil ik dat gegevens van meerdere koppelingen samen te voegen in één overzichtelijk dashboard.	20

Moscow

Vervolgens zijn deze user stories in een moscow diagram gezet om er een prioriteit aan te koppelen zodat duidelijk is welke functionaliteit ik eerst ga voltooien.

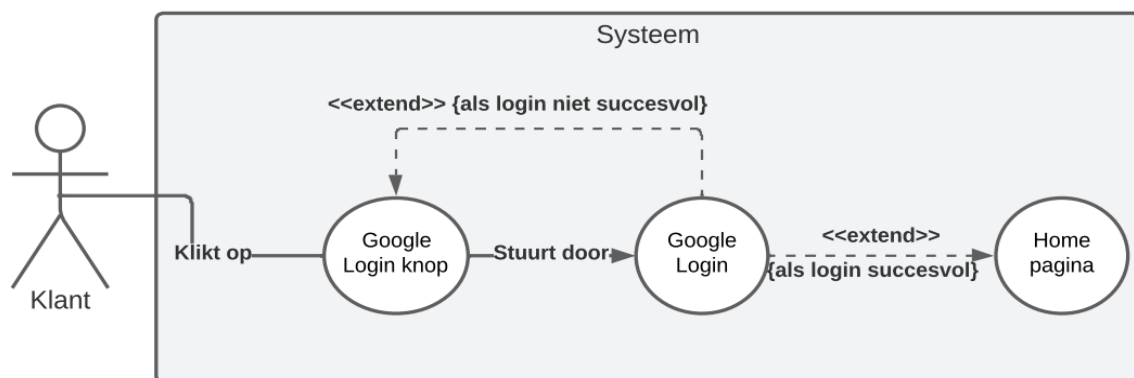
Must have	Should have	Could have	Won't have
Als deelnemer moet ik mijn google fit kunnen koppelen aan de app	Als fysiotherapeut wil ik gezondheidsdata kunnen inzien via een dashboard.	Als fysiotherapeut wil ik meerdere grafiek opties bij het weergeven van klant data	Als fysiotherapeut kan ik andere data inzien van Health Connect
Als deelnemer moet ik mijn apple gezondheid kunnen koppelen aan de app op ios			Als fysiotherapeut kan ik andere data inzien van Google Fit
Als deelnemer moet ik mijn health connect kunnen koppelen aan de app op android.			Als fysiotherapeut kan ik andere data inzien van Apple health
Als fysiotherapeut wil ik de dagelijkse stappen kunnen inzien			
Als fysiotherapeut wil ik de afstand afgelegd door een deelnemer kunnen inzien.			
Als deelnemer moet toestemming kunnen geven om mijn gegevens te delen.			

Use Cases

De algemene functionaliteiten zoals koppeling van externe api, ophalen van data, toestemming geven voor data, weergave opties van data zijn use cases van gemaakt zodat in beeld word gebracht hoe de deelnemer hier mee in contact word gebracht en hoe het systeem werkt.

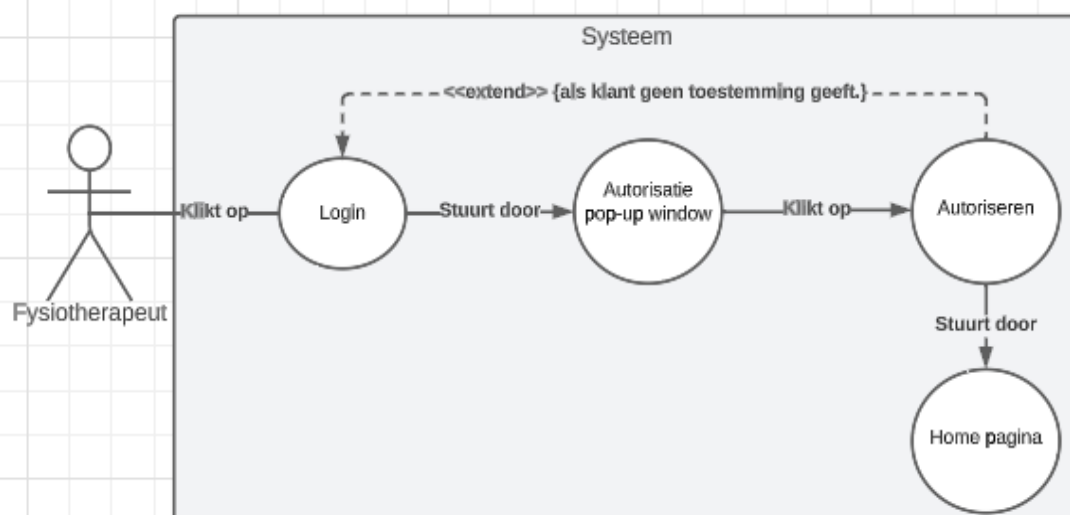
Als deelnemer moet ik kunnen inloggen via Google.

Acteur	Klant
Voorwaarden	Klant is op de login pagina
Triggers	Klant klikt op google login
Hoofd succes scenario	Klant klikt op google login, Klant word gebracht naar externe login van google en logt in.



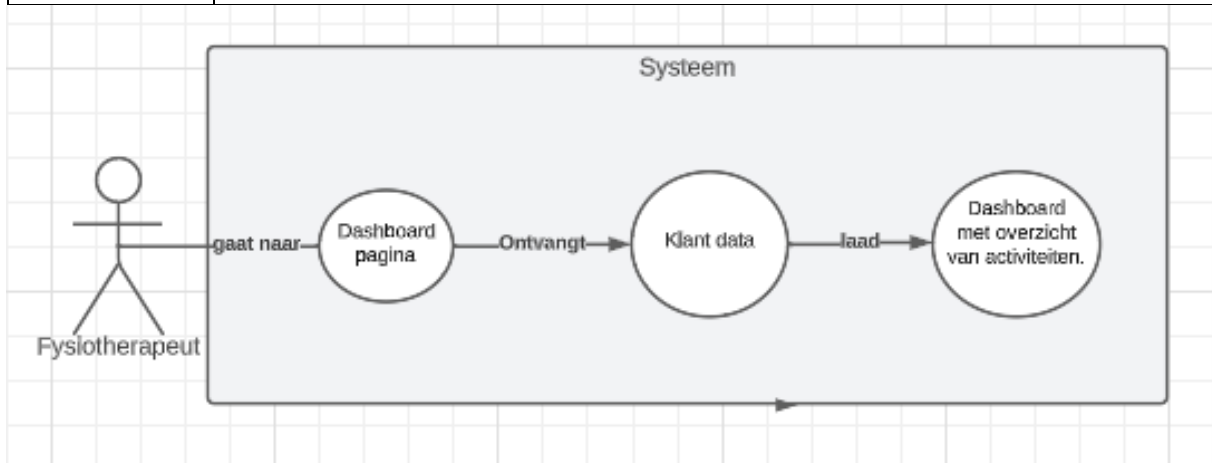
Als deelnemer moet ik toestemming kunnen geven om mijn gegevens te delen.

Acteur	Klant
Voorwaarden	Klant heeft google login ingevoerd
Triggers	Klant logt in met google
Hoofd succes scenario	Klant logt in met google, klant word doorgestuurd naar pop-up window met alle data waar de applicatie naar vraagt, Klant klikt op autoriseren, klant gaat naar home pagina.



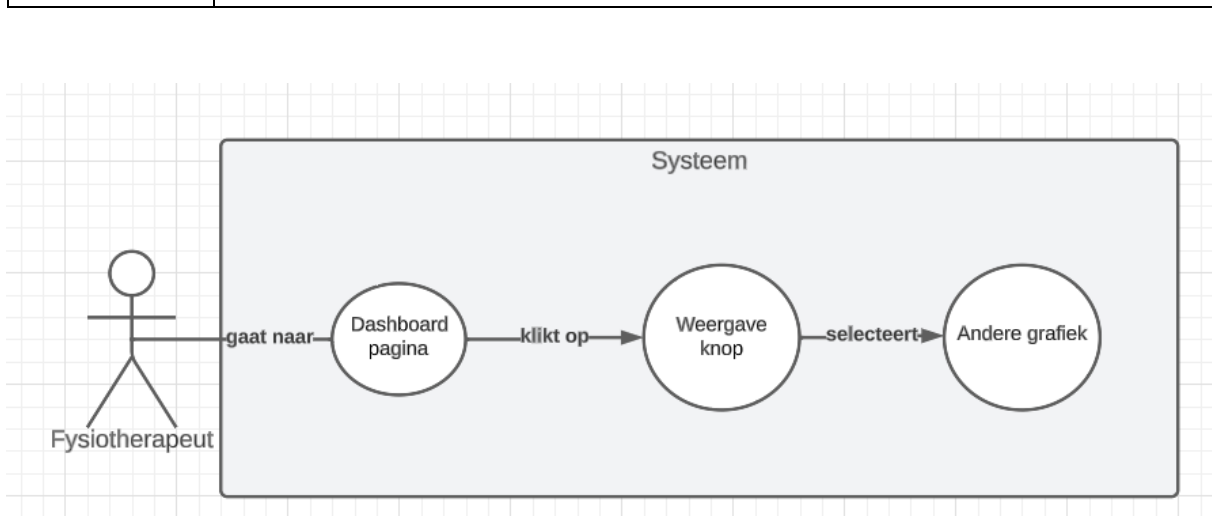
Als fysiotherapeut wil ik gezondheidsdata kunnen inzien via een dashboard.

Acteur	Fysiotherapeut
Voorwaarden	Fysiotherapeut is ingelogd, een of meer klanten die autorisatie hebben gegeven.
Triggers	Fysiotherapeut is op dashboard pagina
Hoofd succes scenario	fysiotherapeut word naar dashboard gebracht, fysiotherapeut ontvangt data, data word weergeven in een dashboard met overzichten van alle activiteiten.



Als fysiotherapeut wil ik meerdere grafiek opties bij het weergeven van klant data.

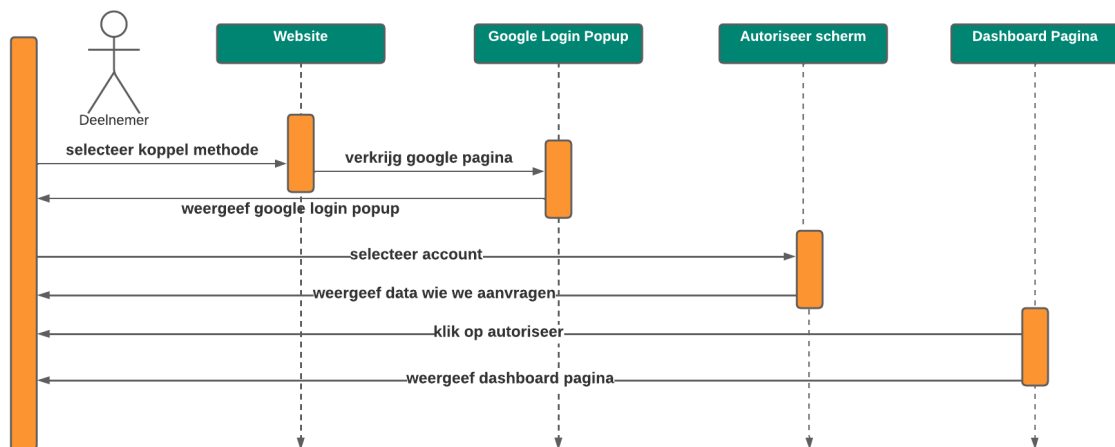
Acteur	Fysiotherapeut
Voorwaarden	Fysiotherapeut is ingelogd, een of meer klanten die autorisatie hebben gegeven.
Triggers	Fysiotherapeut is op dashboard pagina
Hoofd succes scenario	Fysiotherapeut is op dashboard pagina, fysiotherapeut klikt op de knop weergave, Fysiotherapeut selecteerd andere grafiek.



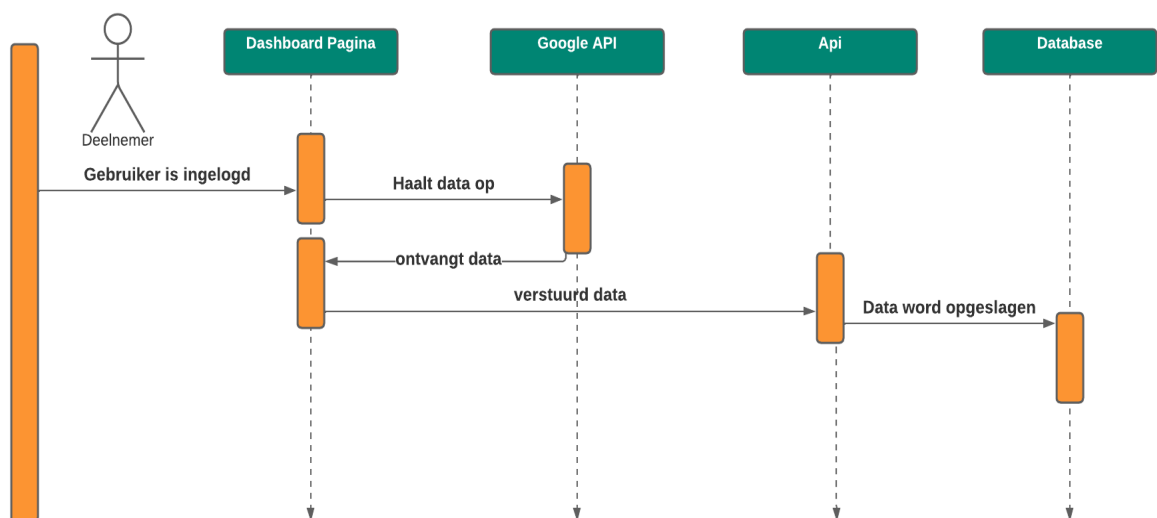
Sequence Diagram

Er zijn sequence diagrammen gemaakt van het verwerken en verzamelen van data en van het toestemming verlenen voor het lezen van data.

Als deelnemer moet ik kunnen inloggen via Google/ Als deelnemer moet ik toestemming kunnen geven om mijn gegevens te delen.



Sequence diagram voor data die verzameld en verwerkt word.(dit is het sequence diagram voor alle user stories die data verzamelen en verwerken.)



Smart Watch diagram directe koppelingen

Voor mijn onderzoek “Welke activiteiten apps hebben de meeste integraties van populaire smartwatches?” is er een diagram gemaakt om goed in te kunnen zien welke gezondheids apps welke smartwatch integraties ondersteunen .

Watches Apps	FIT BIT	SAMSUN G	APPLE	GARMIN	POLAR	AMAZFIT	FOSSIL	MOBVOI
Apple health (IOS)	X	V	V	V	V	V	V	V
Google fit (BOTH)	X	X	V	X	V	V	V	V
Samsung health (BOTH)	X	V	V	X	X	X	X	X
Health Connect (ANDROID)	V	V	X	X	X	X	X	X
Health Connect ; Health tracker (IOS)	X	X	V	X	X	X	X	X
Fit Bit app (BOTH)	V	X	X	X	X	X	X	X
Garmin connect (BOTH)	X	X	V	V	X	X	X	X
Strava (BOTH)	V	V	V	V	V	V	V	V
Polar flow (BOTH)	X	X	V	X	V	X	X	X
Zepp (BOTH)	X	X	V	X	X	V	X	X
Fossil Smartwachtes (BOTH)	X	X	V	X	X	X	V	X
Mobvoi App (BOTH)	X	X	V	X	X	X	X	V
Healthsync (ANDROID)	V	V	X	V	V	X	X	X

Meest Optimale implementatie

Door het vorige diagram heb ik samen met Roy een overzicht kunnen maken van de meest optimale implimentatie voor het Move4Vitality systeem, we hebben alles kunnen dekken behalve fit bit integratie voor IOS.

Watches	FIT BIT	SAMSU NG	APPLE	GARMIN	POLAR	AMAZFI T	FOSSIL	MOBVOI
Apps								
IOS(Apple health)	X	V	V	V	V	V	V	V
ANDROID(Google Fit/Health Connect/Garmin Connections)	V	V	V	V	V	V	V	V

Voor & Nadelen

Bij het maken van de meest optimale implimentatie heb ik de apps die de meeste integratie hebben vergeleken met elkaar hiervoor heb ik voor en nadelen van deze apps verzameld.

Google Fit

Voordelen

- Gratis developer account
- Gratis voor de gebruiker
- Beschikbaar op IOS en Android

Nadelen

- Geen directe verbinding met garmin
- Geen directe verbinding met samsung
- Geen directe verbinding met fitbit

Apple Health

Voordelen

- Integratie met vrijwel alle horloges
- Gratis voor de gebruiker
- Integratie met heelveel andere fitness apps

Nadelen

- Alleen beschikbaar op IOS
- Betaald developer account
- Geen directe verbinding met FitBit.

Strava

Voordelen

- Integratie met alle smart watches.
- Gratis voor developer.
- Beschikbaar op IOS & ANDROID

Nadelen

- Betaald lidmaatschap voor gebruikers.

Health Connect

Voordelen

- Directe integratie met Fitbit, Samsung en google fit en hierdoor ook indirect met alle andere smart watches.
- Gratis voor de gebruiker
- Gratis voor de developer
- Integratie met heel veel andere fitness apps

Nadelen

- Android nodig om te kunnen testen.