

september 2022
t/m
januari 2023

Advies Rapport

Project 'Gebarentaal'

Maartje van Driesten
Wally van de Pas
Piotr Tadrała
Ruben Ulfman

in opdracht van
Prowise B.V.

versie
4 (30-12-22)

INHOUD

1	Samenvatting	3
2	Context	4
3	Onderzoek	5
3.1.	Gebruikersonderzoek	5
3.2.	Literatuuronderzoek	5
3.3.	Veldonderzoek – Prototyping	8
4	Requirements analyse	10
5	Architectuur oplossing	11
6	Dataset en Back-end Logica	12
6.1	Dataset	12
6.2	Back-end Logica	12
7	Front-end Toepassingen	13
8	Voortzetting van het Project	14
	Bronnen en literatuur	15
	Bijlagen	15

DOCUMENT GESCHIEDENIS

Versie	Datum	Wijzigingen
4	30-12-22	<ul style="list-style-type: none"> Verwerking feedback Lennart
3	21-12-22	<ul style="list-style-type: none"> Paragraaf met samenvatting van de verschillende opgestelde adviezen. Paragraaf met conclusie. Aanvulling Piotr op Hand-Tracking.
2	20-12-22	<ul style="list-style-type: none"> Pre-Release review in groep
1	13-12-22	<ul style="list-style-type: none"> Ruben Ulfman <ul style="list-style-type: none"> ✓ Uitbreiding '3.1 Gebruikersonderzoek' ✓ Uitbreiding '4 Requirements analyse' ✓ Uitbreiding '5 Architectuur Oplossing'
	16-12-22	<ul style="list-style-type: none"> Wally van de Pas <ul style="list-style-type: none"> ✓ Uitbreiding '3.2 Literatuuronderzoek' ✓ Uitbreiding '6, 6.1, 6.2 Dataset en back-end logica' ✓ Uitbreiding '8 Voortzetting van het Project'
	16-12-22	<ul style="list-style-type: none"> Maartje van Driesten <ul style="list-style-type: none"> ✓ Uitbreiding '3.2 Literatuuronderzoek' ✓ Uitbreiding '1 Samenvatting'
	17-12-22	<ul style="list-style-type: none"> Ruben Ulfman <ul style="list-style-type: none"> ✓ Uitbreiding '2 Context' ✓ Revisie (zware inkorting) hoofdstukken 3.1, 4 en 5. ✓ Uitbreiding '7 Front-End toepassingen' ✓ Uitbreiding '3.2 Literatuuronderzoek' ✓ Uitbreiding '3.3 Veldonderzoek'
	18-12-22	<ul style="list-style-type: none"> Piotr Tadrała <ul style="list-style-type: none"> ✓ Uitbreiding '3.3 Veldonderzoek' ✓ Uitbreiding '6.1 & 6.2 Dataset en Back-end logica'
	20-12-22	<ul style="list-style-type: none"> Ruben Ulfman <ul style="list-style-type: none"> ✓ Herschrijving '4 Requirements analyse' a.d.h.v. feedback van Lennart op Portfolio versie van bijlage 1.

1 SAMENVATTING

Bijdrage door	: Maartje van Driesten	16-12-22	Ruben Ulfman	17-12-22	Wally van de Pas	21-12-22
Review door	: Ruben Ulfman	17-12-22	Wally van de Pas	21-12-22		

Dit afgelopen semester (30 augustus 2022 t/m 16 januari 2023) zijn wij als student-projectgroep vanuit Fontys Hogeschool aan de slag gegaan met de wens van Prowise om vanuit een inclusiviteit-gedachte onderzoek te doen naar mogelijke functionaliteit om op hun digitale schoolborden automatisch gebarentaal te herkennen en te vertalen. Het is niet alleen handig voor bijscholing van het desbetreffende slechthorende of dove kind om gebarentaal beter te beheersen, maar ook voor de klasgenootjes van het kind.

De projectgroep waarin wij hiermee bezig zijn gegaan bestaat uit Wally van de Pas, Maartje van Driesten, Piotr Tadrata en Ruben Ulfman. Hierbij heeft Wally van de Pas, als werknemer bij Prowise en aandrager van de challenge bij Fontys, de rol als aanspreekpunt dan wel key-user op zich genomen.

Vanuit de wens hebben wij een eenvoudige set eisen opgesteld, welke zijn gevalideerd door Prowise. Aan de hand van deze set eisen hebben wij verkennend onderzoek gedaan naar de verschillende mogelijkheden in technologie, bestaande toepassingen en specifieke eisen van de gebruikers. Om de haalbaarheid van de mogelijkheden te toetsen, hebben wij deze geanalyseerd middels het opstellen van verschillende prototypes waaronder twee mogelijkheden voor back-end logica, verschillende opties voor het opzetten van een dataset gebaren, en een verkenning van de vormgeving en mogelijke inhoud van front-end toepassingen van de functionaliteit.

Op basis van dit uitgebreide vooronderzoek hebben wij onder andere een set requirements opgesteld waaraan toekomstige oplossingen zouden moeten voldoen, en hebben wij adviezen opgesteld over verschillende onderdelen van de functionaliteit. Hierin hebben wij rekening gehouden met het opzetten van de verschillende systeem lagen zodat er later met de bestaande tools van Prowise gecommuniceerd kan worden. Ook is er gekeken naar het vergaren en processen van de training data voor de logica van de applicatie zodat wij een duidelijk beeld kunnen schetsen over de werkzaamheden, kosten of andere middelen die hiervoor nodig zijn.

Als voorzet voor een eventueel vervolgstappen die Prowise met dit rapport kan nemen is ook gekeken naar enkele front-end oplossingen die met deze technologie gemaakt zouden kunnen worden.

Wat betreft een voortzetting als challenge in een tweede semester zijn er een aantal punten naar voren gekomen die nodig zijn om dit haalbaar te maken.

Aan de hand van de analyse naar gebarentaal, en technische prototypen voor herkenning hiervan, is een belangrijke conclusie dat de haalbaarheid van het herkennen van de volledige gebarentaal zeer onrealistisch is. De complexiteit van de gebarentaal is simpelweg te groot, en bestaande technieken en tools zijn hiervoor niet toereikend. Een beperkte herkenning van het alfabet en cijfers zoals gewenst door Prowise, is waarschijnlijk het hoogst haalbare binnen het project. Verdere herkenning vraagt grote investering van tijd in ontwikkeling van technieken.

De volledige documentatie van al het gedane onderzoek is voor de geïnteresseerde lezer terug te vinden in de diverse bijlagen, en geciteerde bronnen in dit document. Zo bevat **Bijlage 1 – Research Gebarentaal** analyse, requirements en advies voorkomend uit de Nederlandse Gebarentaal. **Bijlage 2 – Analyse & Design Front-End** bevat analyse en design m.b.t ontwerpen van front-end applicaties voor de primaire doelgroep.

Met dit onderzoek denken wij een goed doordachte basis voor het maken van een applicatie te presenteren. In dit document kunnen de diverse afdelingen en directie-lagen binnen Prowise informatie vinden over de mogelijkheden van de mogelijke technologieën en de eisen die het herkennen van Nederlands gebarentaal met zich meebrengt. Hierin is rekening gehouden met de schaalbaarheid en implementatie binnen de bestaande tools van Prowise en natuurlijk met de eisen van Prowise op het gebied van veiligheid. Door dit onderzoek is het voorafgaande aan de eventuele development duidelijk wat een applicatie wel kan waarmaken en wat niet, en welke middelen hiervoor nodig zijn.

Graag zouden wij dan ook dit onderzoek gebruiken als startpunt van ons nieuwe semester en zo een daadwerkelijke applicatie willen opleveren.

2 CONTEXT

Bijdrage door : Ruben Ulfman 17-12-22

Review door : Wally van de Pas 21-12-22

Prowise levert onder andere smartboards en personal devices voor het basisonderwijs. Vanuit een inclusiviteit-gedachte is Prowise geïnteresseerd in het verkennen of de devices mogelijk in de toekomst middels hun camera in staat zijn de Nederlandse Gebarentaal te detecteren om zo doven en slechthorenden beter te kunnen bedienen in de klas op de basisschool.

Door Prowise is gevraagd om (prototype) functionaliteit die in eerste instantie in staat is tot het herkennen van het Nederlandse gebarenalfabet en de cijfers in het Nederlandse Gebarentaal. Bij doorvraag over de front-end toepassingen heeft Prowise aangegeven dat we bijvoorbeeld aan serieuze les-toepassingen kunnen denken om gebaren aan te leren via een quiz-module, maar ook spelende manieren zoals een spelletje Galgje. Wel geven ze daar direct bij aan dat galgje zelf niet gewenst is – maar meer een voorbeeld van op een spelende manier woorden en letters leren.

Ook heeft Prowise aangegeven dat het bijvoorbeeld niet gewenst is dat functionaliteit een internetverbinding vereist, en dat functionaliteit voor het herkennen en vertalen van gebaren los beschikbaar moet zijn van functionaliteit waarin deze herkenning en vertaling gebruikt wordt.

Voor meer details m.b.t. context, verwijzen we naar [Bijlage 2 – Analyse & Design Front-End](#).

3 ONDERZOEK

3.1. GEBRUIKERSONDERZOEK

Bijdrage door : Ruben Ulfman 13-12-22

Review door : Wally van de Pas 21-12-22

Voor meer details, verwijzen we naar **Bijlage 1 – Research Gebarentaal**.

LEERLINGEN IN HET BASISONDERWIJS

Er zijn in Nederland 1,5 miljoen doven of slechthorenden (1), zijn er daarvan ongeveer 500.000 doven, waarvan er zo'n 30.000 Nederlanders afhankelijk zijn van de Nederlandse Gebarentaal. (2)(3) Van de Nederlandse bevolking waren er op 1 januari van de totaal 17.6 miljoen inwoners zo'n 1.4 miljoen kinderen (4) van basisschool leeftijd (4 tot 12 jaar). Dat is afgerond zo'n 10%. Dat zou betekenen dat we in Nederland ongeveer 3000 dove kinderen in onze doelgroep afhankelijk zouden hebben van Nederlandse Gebarentaal, met tussen de 50.000 en 150.000 extra slechthorenden. Leeftijdsspecifieke hoeveelheden zijn niet realistisch, omdat bijvoorbeeld veel mensen op latere leeftijd door gehoorbeschadiging toetreden tot de ~1.5 miljoen doven of slechthorenden in Nederland. De 3000 dove kinderen (en aanvullende slechthorenden) is slechts een indicatie.

De gebruikersgroep bestaat niet alleen uit dove/slechthorende kinderen, maar als deze doelgroep deelneemt aan regulier onderwijs, dan juist ook uit hun klasgenootjes. Door de klasgenootjes beter bekend te maken met gebarentaal, worden de dove/slechthorende leerlingen beter meegenomen in de klasdynamiek.

Binnen het spectrum van 6 tot 12 jaar verschillen de behoeften sterk. In het onderste deel van het spectrum is het leren lezen en schrijven, de fijne motoriek en hand-oog coördinatie nog in ontwikkeling. In het bovenste deel van het spectrum beginnen de kinderen al te puberen.

LERAREN IN HET BASISONDERWIJS

De tweede groep gebruikers zijn de leraren in het basisonderwijs. Tijdens hun lessen waarbij ze gebruik maken van Prowise schoolborden, zouden ze ook de gebarentaal applicaties kunnen gebruiken. Deze groep gebruikers is voornamelijk begeleiding van de hierboven opgenomen gebruikersgroep: de leerlingen in het basisonderwijs.

BRONNEN DIT HOOFDSTUK

- 1 NOS. (2019, 16 oktober). *Doven en slechthorenden minder gezond door gebrekkige medische communicatie*. NOS.nl. Geraadpleegd op 21 oktober 2022, van <https://nos.nl/artikel/2306256-doven-en-slechthorenden-minder-gezond-door-gebrekkige-medische-communicatie>
- 2 Lammertink, I. (2019, 21 mei). *Iedereen heeft baat bij kennis van Nederlandse Gebarentaal*. Wetenschap.nu. Geraadpleegd op 21 oktober 2022, van <https://wetenschap.nu/iedereen-heeft-baat-bij-kennis-van-nederlandse-gebarentaal/>
- 3 *Nederlands Gebarentaalcentrum*. (z.d.). Geraadpleegd op 21 oktober 2022, van <https://www.gebarentaalcentrum.nl/vragen+over+NGT>
- 4 Centraal Bureau voor de Statistiek. (2022, 30 september). *Jongeren*. Geraadpleegd op 11 november 2022, van <https://www.cbs.nl/nl-nl/visualisaties/dashboard-bevolking/leeftijd/jongeren>

3.2. LITERATUURONDERZOEK

Bijdrage door : Maartje van Driesten 16-12-22 Wally van de Pas 16-12-22 Ruben Ulfman 16-12-22

Review door : Ruben Ulfman 16-12-22

COMPUTER VISION

Computer Vision (CV) en Machine Learning (ML) worden vaak onder dezelfde noemer vermeld. Toch is het voor onze test applicaties beter om deze los van elkaar te zien om zo de verschillende ML onderdelen beter te differentiëren. Voor de oplossing was CV zowel nodig voor het aanleren van het model als voor het uiteindelijke gebruik van de applicatie. De bibliotheek (2) die hiervoor gebruikt wordt is die van Media Pipe. Dit is een open source platform dat door google is gecreëerd om Computer Vision makkelijk aan projecten toe te voegen. Het onderzoek dat wij hierna hebben gedaan is voornamelijk gebaseerd op de GIT van Media Pipe.

Een alternatief of aanvulling op MediaPipe is OpenCV⁽⁶⁾. Dit is een enorme open-sourcebibliotheek voor computervisie, machine learning en beeldverwerking, en speelt nu een belangrijke rol in realtime (ver)werking, wat erg belangrijk is in de systemen van vandaag. Door het te gebruiken, kan men afbeeldingen en video's verwerken om objecten, gezichten of zelfs handschrift van een mens te identificeren. Wanneer het is geïntegreerd met verschillende bibliotheken, zoals NumPy, kan python de OpenCV-arraystructuur voor analyse verwerken. Om beeldpatronen en hun verschillende kenmerken te identificeren, gebruiken we vectorruimte en voeren we wiskundige bewerkingen uit op deze kenmerken.

Voor een uitgebreide toelichting willen we verwijzen naar [Bijlage 4 – OpenCV-Python](#).

MACHINE LEARNING

Om de met Media Pipe opgehaalde data te gebruiken zijn we opzoek gegaan naar een manier om de data om te zetten naar een Machine Learning (ML) model. In Python was hiervoor de bibliotheek van TensorFlow ⁽³⁾ beschikbaar. Hiermee kun je op een relatief makkelijke manier ML pijplijnen opzetten en ML modellen aanleren.

Omdat dit onderwerp te interessant was om alleen te focussen op de mogelijkheden van TensorFlow hebben we de Data Science lectures van Steven Brunton ⁽⁴⁾ bekeken en deze vertaald in een guide voor gebruik in dit project. In een van de video's wordt uitgebreid aandacht besteed aan LSTM layers. Om ook hier weer een verdieping in te maken hebben we op de site van datascience.eu een mooi stuk gevonden. ⁽⁵⁾

(NEDERLANDSE) GEBARENTAAL

Een specifiek kenmerk van een deel van de tweede gebruikersgroep is dat hier gaat om dove en slechthorende kinderen. Deze kinderen zijn gedeeltelijk, of volledig, afhankelijk van de Nederlandse Gebarentaal voor hun communicatie. Om die reden is het belangrijk onderzoek te doen naar de gebarentaal zelf. De taal is uitgebreid toegelicht in [Bijlage 1 – Research Gebarentaal](#), maar omdat de Nederlandse Gebarentaal een volwaardig eigen taal is, kan het nooit volledig dekkend worden beschreven.

De belangrijkste conclusie die we willen trekken zijn dat het zeer waarschijnlijk niet binnen de mogelijkheden valt om de volledige gebarentaal te gaan herkennen. De complexiteit van de taal maakt dit simpelweg niet haalbaar. De taal heeft bijvoorbeeld ruimtelijk gebruik die de context geeft waarvan de vertaling van een gebaar afhankelijk is. Een ander voorbeeld van complexiteit is gebruik van mimiek die een gebaar verandert van bevestigend in ontkennend in betekenis.

Dat gezegd hebbend, is beperkte herkenning van gebaren een goede fundering, en zal ook al een goede bijdrage leveren aan de wens van Prowise. Bij beperkte herkenning denken wij aan het gebarenalfabet en -cijfers. Naarmate de beschikbare data, en bekendheid met de taal en systemen, groeit, zal mogelijk verdere herkenning van gebarentaal ook tot de mogelijkheden behoren.

Bovenstaande conclusie is gebaseerd op uitgebreid onderzoek naar Nederlandse Gebarentaal waarbij gebruik is gemaakt van meerdere bronnen. Voor meer details, zie [Bijlage 1 – Research Gebarentaal](#).

DIDACTIEK

Door te begrijpen hoe kinderen leren helpt het ons om betere software te maken. Door de beleevingswereld in lerende vorm van een kind te onderzoeken, helpt het ons de fasen van de applicatie te maken met bijvoorbeeld als doel; Een kind iets op een goede manier te laten motiveren zodat het kind hiermee de juiste informatie in het lange termijn geheugen onthoudt.

Om te begrijpen hoe kinderen leren, is het van belang om verschillende fasen in het te onderscheiden:

- De motivatiefase. De leerling moet gemotiveerd raken om te leren.
- De opmerkzaamheidsfase. De leerling richt zijn aandacht op dat wat hij wil leren.
- De opnamefase. De leerling probeert de leerstof in zich op te nemen.
- De geheugenfase. Het geleerde wordt opgeslagen in het geheugen.
- De herinneringsfase. Het terughalen van het geleerde.
- De generalisatiefase. Het geleerde kunnen toepassen in andere situaties.
- De uitvoeringsfase. De leerling laat zien wat hij geleerd heeft.

Voor een uitgebreide toelichting willen we verwijzen naar **Bijlage 3 – UX-UI design**. Dit document bevat een gedetailleerd gebruikersonderzoek met als onderzoekersvraag ‘Welke zaken zijn nodig te weten om de applicatie goed bij de belevingswereld van het kind te laten passen? Uit deze onderzoekersvraag komt een advies.

BRONNEN DIT HOOFDSTUK

- 1 *Nederlands Gebarencentrum*. (z.d.). <https://www.gebarencentrum.nl/vragen+over+NGT>
- 2 *Hands*. (z.d.). mediapipe. <https://google.github.io/mediapipe/solutions/hands.html>
- 3 *Machine learning education* |. (z.d.). TensorFlow. <https://www.tensorflow.org/resources/learn-ml>
- 4 Brunton, S. (2019, 5 juni). *Intro to Data Science: Overview*. YouTube. <https://www.youtube.com/watch?v=pzo13OPXZS4&feature=youtu.be>
- 5 <https://datascience.eu/nl/machine-learning/inzicht-in-lstm-netwerken/>
- 6 *OpenCV open-source computer vision*. (z.d.). Geraadpleegd op 14 oktober 2022, van <https://docs.opencv.org/4.6.0/d1/dfb/intro.html>
- 7 *Tips van een ervaren lerares uit de praktijk*. (z.d.). Geraadpleegd op 3 december 2022, van <https://wij-leren.nl/hoe-kinderen-leren.php>

3.3. VELDONDERZOEK – PROTOTYPING

Bijdrage door	: Wally van de Pas	16-12-22	Piotr Tadrała	18-12-22	Ruben Ulfman	17-12-22
Review door	: Ruben Ulfman	16-12-22	Wally van de Pas	21-12-22		

In ons onderzoek naar de verschillende mogelijkheden kwamen wij vooral op de splitsing tussen een oplossing met en zonder Machine Learning. Ook hebben wij onderzoek gedaan naar het scheiden van back-end logica en front-end gebruik hiervan, via het gebruik maken van een (lokale) API.

MACHINE LEARNING

De eerste oplossing is door ons verder onderzocht. Als basis voor het onderzoek is vooral gekeken naar een aantal YouTube tutorials (1) die gebruik maken van Mediapipe en TensorFlow om zo de key-frames te traceren en deze met Machine Learning (ML) om te zetten naar een getraind ML model. In zijn prototype zijn wij bezig geweest om de ML (LSTM) layers en andere parameters te finetunen voor gebruik met handgebaren. Helaas kwam hij tot de conclusie dat hoe groter de dataset (hoe meer gebaren), hoe moeilijker het systeem de keuzes kon bevestigen.

STATISCHE DATA

De Fingerpose-oplossing maakt gebruik van statische gegevens om handgebaren te herkennen. Dankzij deze aanpak is het mogelijk om het Fingerpose-algoritme snel nieuwe data aan te leren.

Fingerpose-datasets zijn verzamelingen van data die worden gebruikt om handgebaren te herkennen en te classificeren. Deze datasets bestaan uit simpele parameters van handen die verschillende gebaren maken, samen met labels die aangeven welk specifiek gebaar dat het is.

In een Fingerpose dataset verwijzen de parameters 'Finger direction', 'Finger curl' en 'Confidence points' naar specifieke kenmerken van de handgebaren. De parameters "Finger direction" en "Finger Curl" kunnen de positie en kromming van de vingers in elke frame beschrijven, terwijl de "Confidence points" aangeeft voor hoe ver dat de parameters overeenkomen met een perfecte match. Deze parameters worden vervolgens gebruikt om de ankerpunten van de webcam-feed te classificeren het gebaar naar een dataset item verwijzen.

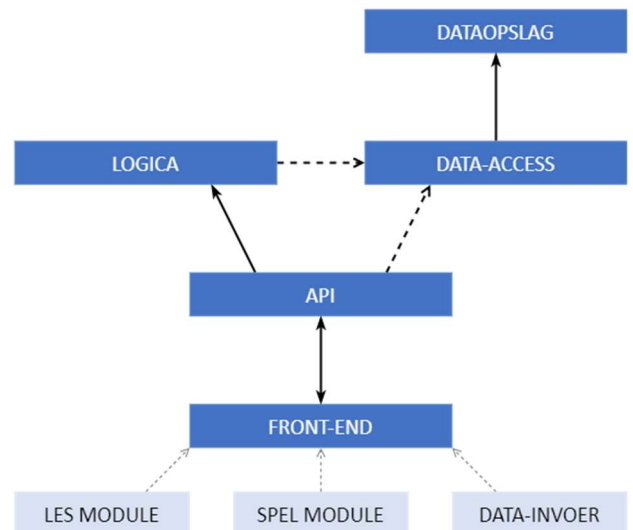
HAND-TRACKING

Mediapipe maakt gebruik van machine learning om handen in real-time te detecteren en te volgen. Het systeem werkt door gebruik te maken van een set ankerpunten, op de handen en vingers om de positie en oriëntatie van de handen te visualiseren. Deze ankerpunten worden gedetecteerd met behulp machine learning-algoritmen die de vorm en beweging van de handen analyseren. Het systeem gebruikt vervolgens deze ankerpunten om de handen te volgen terwijl ze bewegen, wat het in staat stelt om de bewegingen van de handen nauwkeurig in real-time te volgen.

Met behulp van deze hand-tracking technologie kan Prowise de devices nog gebruiksvriendelijker maken door middel van handgebarenbesturing. Bijvoorbeeld, in plaats van een knop in te drukken om een bepaalde actie uit te voeren, kan de gebruiker simpelweg een specifiek handgebaar maken om hetzelfde resultaat te bereiken. Dit kan het bedienen van het apparaat voor sommige gebruikers vereenvoudigen en het gebruik ervan nog aangenamer maken.

API

Om het mogelijk te maken dat het back-end (*logica en data-access in schematische weergave hiernaast*) in een andere programmeertaal opgezet is dan het front-end, en het back-end door meerdere front-ends (her)gebruikt kan worden, hebben we ook onderzoek gedaan naar API's, zowel online API's waarbij het back-end gehost zou worden als webservice, en lokale API's waarbij het back-end zijn eigen dedicated applicatie en proces heeft. Hiervoor heeft Ruben een prototype ontwikkeld waarbij gebruik gemaakt wordt van sockets om informatie uit te wisselen tussen 2 applicaties (een proces wat ook wel InterProces Communicatie (IPC) wordt genoemd). Het prototype bestaat uit 2 endpoints: een endpoint utility gecodeerd in c# wat geïntegreerd kan worden in een c#/Unity applicatie, en een endpoint gecodeerd in Python, wat makkelijk geïntegreerd kan worden in een python applicatie, en waarmee back-end logica geëxposeerd kan worden voor gebruik in het front-end.



Zie ook Hoofdstuk 5 in dit document.

Belangrijke conclusie hieruit is dat het niet alleen een goede oplossing zou zijn, maar ook relatief eenvoudig uit te breiden zou zijn naar dedicated machines voor de processen, die lokaal draaien in een netwerk zonder internetverbinding. Je zou dan het positieve aspect hebben van een webservice voor de logica – namelijk dat de user-interface-machine hier geen werkkraft voor nodig heeft, zonder dat je een internetverbinding nodig hebt.

BRONNEN DIT HOOFDSTUK

- 1 Renotte, N. (2021, 18 juni). *Sign Language Detection using ACTION RECOGNITION with Python | LSTM Deep Learning Model*. YouTube. <https://www.youtube.com/watch?v=doDUihpi6ro&feature=youtu.be>
- 2 Schallwig, A. (2022, 15 november). Rock, Paper, Scissors with hand gesture recognition
Medium: <https://medium.com/geekculture/rock-paper-scissors-with-hand-gesture-recognition-841297a7d915>

4 REQUIREMENTS ANALYSE

Bijdrage door : Maartje van Driesten 16-12-22 Ruben Ulfman 13-12-22

Review door : Wally van de Pas 21-12-22

EISEN VAN STAKEHOLDER PROWISE

- ✓ De applicatie moet het gebarenalfabet en de cijfers van de Nederlandse Gebarentaal kunnen vertalen
- ✓ Buiten het alfabet en cijfers, moet de applicatie ook enkele woordgebaren kunnen herkennen zoals 'Hallo'
- ✓ Bij gebruik van de applicatie moet geen internetverbinding benodigd zijn
- ✓ De functionaliteit voor het herkennen en vertalen van gebaren moet los beschikbaar zijn van eventuele front-end applicaties
- ✓ De applicatie moet functioneren op een Prowise digitaal schoolbord
- ✓ De uiteindelijke applicatie is bedoeld voor gebruik in het basisonderwijs
- ✓ De applicatie moet gebaren zowel gebaren gemaakt door volwassenen, als door kinderen herkennen

REQUIREMENTS

Op basis van het gedane onderzoek, en de eisen van de stakeholder Prowise, willen wij de volgende items User-requirements opstellen voor het uiteindelijke softwaresysteem. Voor het volledige onderzoek waarop deze requirements gebaseerd zijn, zie [Bijlage 1 – Research Gebarentaal](#).

De requirements zijn geprioriteerd volgens MoSCoW, dit houdt in van hoogste naar laagste prioriteit:

Must have → **Should have** → **Could have** → **Won't have**.

URS	Prio	Bron	Omschrijving
1	Must	Prowise	Het Softwaresysteem moet gebaren (minimaal handalfabet en gebarencijfers) kunnen herkennen .
1.A	Must	Prowise	Het systeem moet zowel gebaren gemaakt door kinderen en gebaren gemaakt door volwassenen kunnen herkennen.
1.B	Must	Analyse	Om het handalfabet en de gebarencijfers te kunnen herkennen zal het softwaresysteem in ieder geval de handvorm, de vingers, en de beweging en oriëntatie van de hand, de handpalm en de vingers moeten kunnen registreren .
1.C	Won't	Analyse	Om woorden te herkennen , moet het systeem het systeem relatieve positie ten opzichte van het lichaam, ruimtelijk gebruik , en non-manuele elementen herkennen.
1.D	Won't	Analyse	Non-manuele elementen die herkent moeten worden (zowel statisch als bewegend) om de volledige gebarentaal te kunnen vertalen zijn hoofd, schouders, ellenboog, pols, wenkbrauwen, ogen, mond en gelaatsuitdrukking .
2	Must	Prowise	De applicatie moet een eenvoudig front-end bevatten om de vertaling van herkende gebaren te kunnen tonen aan de gebruiker.
2.A	Should	Prowise	Om toepassing van de functionaliteit te demonstreren, zou een tweetal prototype front-end opgezet kunnen worden. Eén van deze front-ends is een les-module , waarin de gebruiker middels bijvoorbeeld eenvoudige quizen gebarentaal leert.
2.B	Should	Prowise	Om toepassing van de functionaliteit te demonstreren, zou een tweetal prototype front-end opgezet kunnen worden. Een tweede van deze front-ends is een spel-module , waarin de gebruiker middels wat eenvoudige letter-spelletjes de gebarenherkenning kan gebruiken.
3	Must	Prowise	De applicatie(s) moeten volledig (logica, dataset, front-end) offline beschikbaar zijn
4	Must	Prowise	De applicatie(s) moeten werkend zijn op een Prowise digitaal schoolbord .
5	Should	Analyse	Om uiteindelijk ook woorden te kunnen herkennen moet het softwaresysteem uitbreidbaar zijn met herkenning voor relatieve positie en non-manuele elementen zoals gezichtsuitdrukking.
6	Could	Analyse	De dataset moet dynamisch uitbreidbaar zijn, of vervangen worden door een alternatieve dataset voor bijvoorbeeld een andere gebarentaal

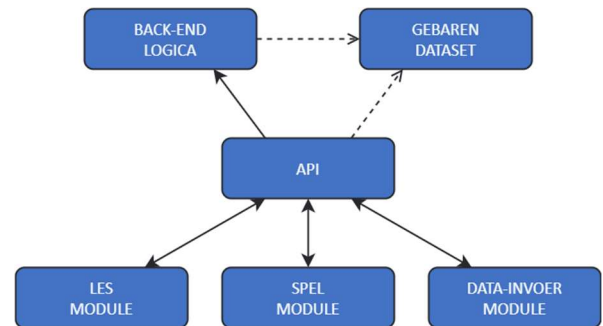
5 ARCHITECTUUR OPLOSSING

Bijdrage door : Ruben Ulfman 13-12-22

Review door : Wally van de Pas 21-12-22

Wij stellen voor om de functionaliteit voor de applicatie op te splitsen in 6 losjes gekoppelde modules of lagen. De ruwe relatie tussen deze lagen is in het schema hiernaast weergegeven

- ✓ **Gebaren Dataset** (Data & Data-access)
- ✓ **Back-End Logica** (Algoritmeik & Logica in Python)
- ✓ **API** (Communicatie tussen Back-End & Front-End)
- ✓ **Les Module** (Front-End in Unity/C#)
- ✓ **Spel Module** (Front-End in Unity/C#)
- ✓ **Data-Invoer Module** (Front-End in Android/iOS – Mobile first)



Door deze lagen los te ontwikkelen en alleen met soft-dependencies te koppelen, zijn de verschillende onderdelen uitwisselbaar. Moet er een nieuwe Front-End komen, of een Front-End module worden vervangen? Geen probleem, de API, back-end en data-set blijven overeind. Moet een andere data-set gebruikt worden? Geen probleem, het front-end, de logica en de API blijven overeind. En zo verder.

KORTE TOELICHTING

Voor een uitgebreide toelichting willen we verwijzen naar [Bijlage 1 – Research Gebarentaal](#). Dit document bevat een gedetailleerd advies voor het opzetten van de architectuur.

Gebaren Dataset

Ons advies voor de dataset is om in eerste instantie gebruik te maken van statische gestructureerde data, zoals een SQL-database of in-code classes. Dit zou uiteindelijk dynamisch kunnen worden uitgebreid naar een semigestructureerde dataset met bijvoorbeeld JSON-documenten. Als uiteindelijk gebruikers ook gebaren kunnen gaan toevoegen aan de dataset is het het overwegen waard om ongestructureerde data te gaan gebruiken in bijvoorbeeld NoSQL databases als MongoDB. Zie ook hoofdstuk 6.1. voor details over de inhoud van de dataset.

Back-End Logica

Ons advies is om deze logica op te zetten in Python. Dit sluit goed aan bij bestaande Computer Vision en Machine Learning tooling, en is uitstekend geschikt voor Dataprocessing en analyse. Zie ook hoofdstuk 6.2. voor meer details over het back-end.

API

Ons advies is om in eerste instantie een locale API op te zetten die via bijvoorbeeld sockets Inter-Proces-Communicatie (IPC) mogelijk maakt. Deze oplossing maakt het mogelijk om aan de eis van Prowise te voldoen om de applicatie volledig offline te kunnen laten draaien, en beid tegelijk de mogelijkheid om zonder lastige integratie schema's de Back-End in Python, en de Front-End in C#/Unity te ontwikkelen.

Met oog op de toekomst en de meer beperkte hardware beschikbaar in de digitale schoolborden, zouden wij echter wel adviseren het back-end via een webservice toegankelijk via een web-API beschikbaar te maken. Dat staat het toe de gebaarherkenning ook te gebruiken op apparaten met beperkte processing power, want het zware denkwerk wordt dan uitgevoerd op een centrale server.

Front-End

Voor de les- en spelmodule kan de bestaande ontwikkelstrategie binnen Prowise gebruikt worden, met de enige uitzondering dat voor de vertaal-logica de (lokale) API aangesproken zou moeten worden.

Voor de data-invoer module adviseren wij een mobile-first oplossing, als het wenselijk is dat de dataset in de toekomst ook door gebruikers uitgebreid kan worden.

Zie ook hoofdstuk 7 voor details over de les- en spelmodule front-ends.

6 DATASET EN BACK-END LOGICA

6.1 DATASET

Bijdrage door	: Wally van de Pas	16-12-22	Piotr Tadrała	18-12-22
Review door	: Ruben Ulfman	16-12-22		

Bij de oplossing van het mediapipe project zit een module om zelf de datasets in te leren. Deze doe je een aantal keer voor aan de RGB-camera en zo worden de keypoints opgenomen en omgezet naar een model. Omdat dit veel tijd in beslag nam zijn we opzoek gegaan naar grote datasets die we kunnen gebruiken. Hierbij zijn we op de site [kaggle.com](https://www.kaggle.com) (1) een grote Duitse dataset gekomen met duizenden voorbeelden van handen. Helaas is het ons niet gelukt om deze dataset werkend te krijgen omdat het algoritme uitgaat van bewegende beelden en de dataset statische handen bevat.

Ook met de aangeleerde dataset is het niet gelukt om een consequente en stabiele tool te maken voor het uitlezen van de gebaren. Ons advies is dan ook om niet voor de mediapipe oplossing, maar voor de fingerpose oplossing te gaan.

Fingerpose maakt gebruik van een statische verzameling die de positie en kromming van elke vinger voor elk gebaar beschrijft. Om een dataset te genereren, moet handmatig een module worden gemaakt die de webcam-feed kan omzetten naar parameters en deze vervolgens toewijzen aan een gebaar. Dit heeft het voordeel dat je heel specifiek kunt aangeven wat de parameters van elk gebaar zijn. Het is ook mogelijk om meerdere dataset-items aan een gebaar toe te wijzen om de nauwkeurigheid te verbeteren.

BRONNEN DIT HOOFDSTUK

- 1 Find Open Datasets and Machine Learning Projects | Kaggle. (z.d.). <https://www.kaggle.com/datasets>
- 2 Prasad, P. FingerposeEstimate python module
Github: <https://github.com/Prasad9/Classify-HandGesturePose/tree/master/pose/utlis>

6.2 BACK-END LOGICA

Bijdrage door	: Wally van de Pas	16-12-22	Piotr Tadrała	18-12-22	Wally van de Pas	30-12-22
Validatie door	: Ruben Ulfman	16-12-22				

De backend van het mediapipe project maakt gebruik van machine learning (ML) om de keypoints die opgenomen worden samen te voegen en om te zetten naar een machine learning model. Voor dit hele proces worden diverse libraries gebruikt zo worden de keypoints opgespoord met mediapipe van google, de keypoints worden vervolgens omgezet naar een ML model doormiddel van TensorFlow. Dit model kan vervolgens weer met mediapipe worden vergeleken met de live input van de camera om te beoordelen of er een letter getoond wordt. De theorie van dit project lijkt uitermate geschikt voor zowel statische als bewegende gebaren. Toch bleek dat de ML modellen niet zo in te regelen zijn dat ze consequente en stabiele output genereren. Het probleem hierin was dat voor het LSTM-model dat werd aangeraden voor tijdslijn-gebonden data zoals video fragmenten de verschillende gebaren niet genoeg van elkaar verschilde. We zouden kunnen overstappen naar een ander model, maar dan zouden we weer in de problemen komen met de gebaren die beweging gebruiken.

De back-end van Fingerpose omvat het gebruik van computervisie algoritmen om de webcam-feed te analyseren en de positie van de vingers van een gebruiker te identificeren. Om dit te bereiken, gebruikt Fingerpose een combinatie van machine learning-technieken en beeldverwerkingsalgoritmen om de webcam-feed in real-time te analyseren. Het zoekt naar patronen en kenmerken in de afbeeldingen die corresponderen met specifieke vingerbewegingen en gebruikt deze informatie om de vingergebaren van de gebruiker nauwkeurig te volgen en te interpreteren.

BRONNEN DIT HOOFDSTUK

- 1 Prasad, P. FingerposeEstimate python module
Github: <https://github.com/Prasad9/Classify-HandGesturePose/tree/master/pose/utlis>
- 2 GOOGLE LLC. Mediapipe Hands
Github.io: <https://google.github.io/mediapipe/solutions/hands.html>

7 FRONT-END TOEPASSINGEN

Bijdrage door : Ruben Ulfman 17-12-22 Wally van de Pas 21-12-22

Review door : Wally van de Pas 21-12-22

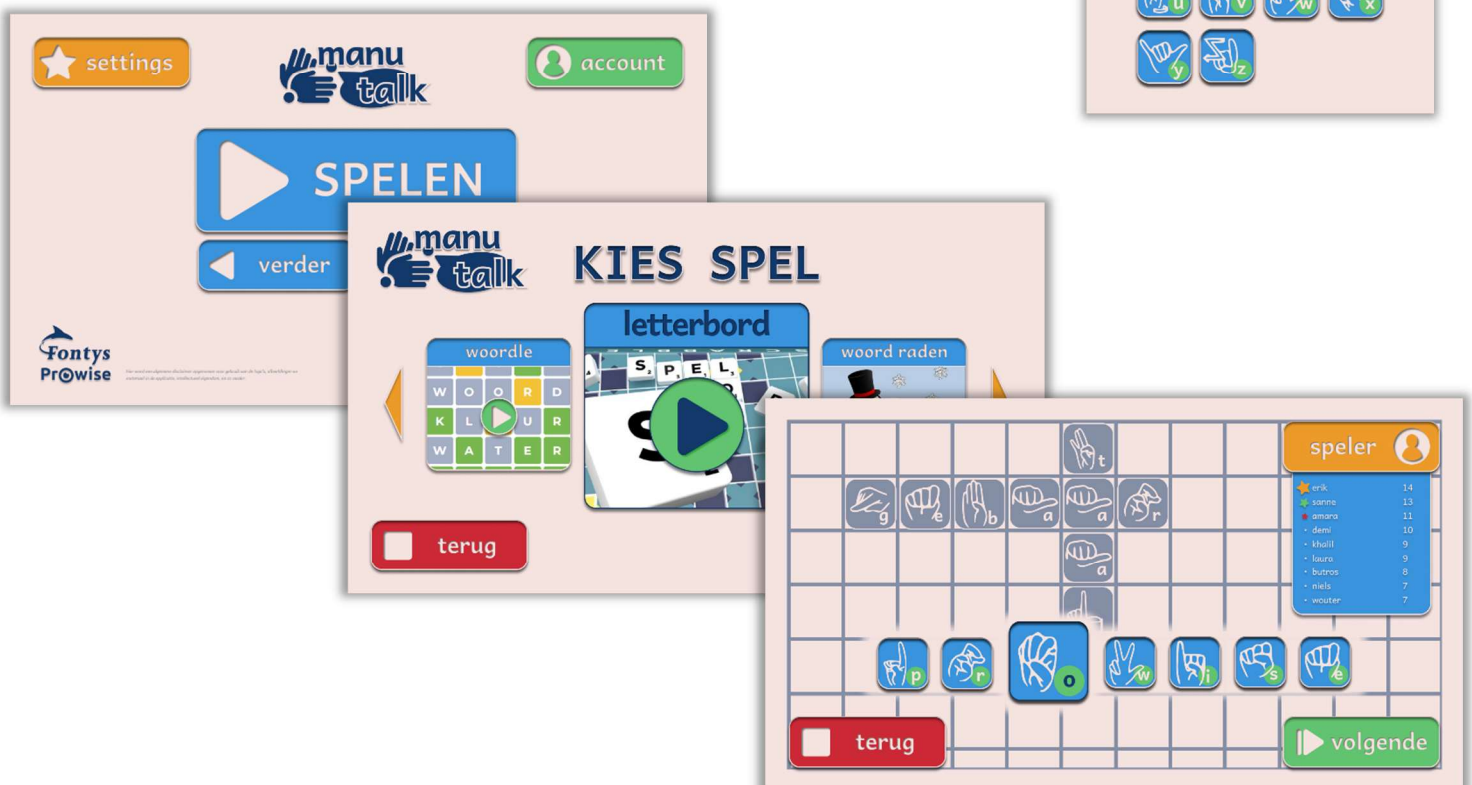
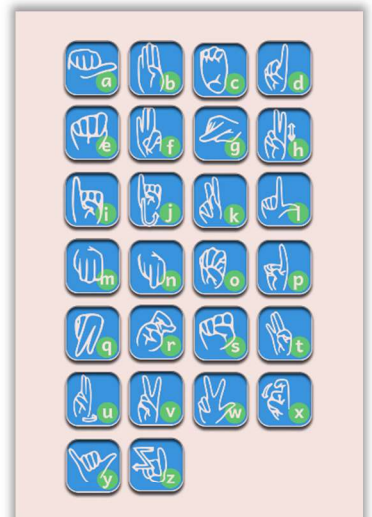
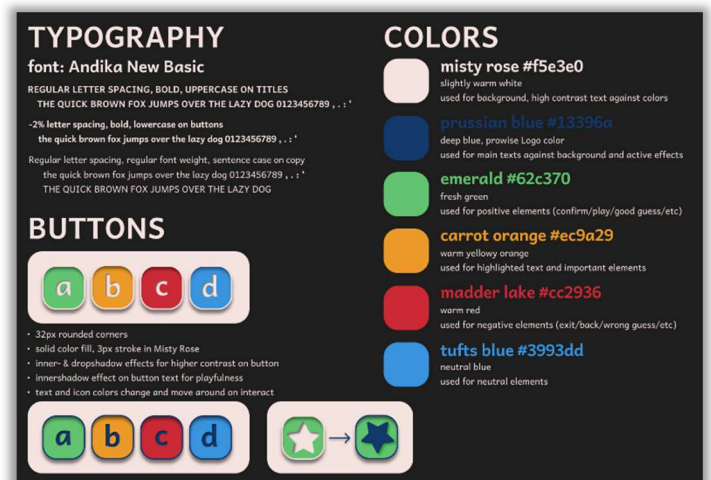
Voor het ontwikkelen van de front-end toepassingen adviseren wij een user-centered designstrategie aan te houden. Hierbij wordt hoofdzakelijk ontwikkeld om aan de eisen van de (primaire) doelgroep te voldoen. In dit project gaat het hoofdzakelijk om basisschoolkinderen.

Hiernaast (boven) is een styleguide opgenomen die is voortgekomen uit onderzoek naar UI/UX voor deze doelgroep. Dit is gebaseerd op de volgende belangrijke conclusies uit het onderzoek:

- Gebruik **primaire kleuren**.
- Gebruik goed leesbare **simplele fonts**
- Gebruik **grotere font**-grootten
- Gebruik weinig of **geen abstractie**
- **Vermijd kopij** zoveel mogelijk.
- Houd **teksten** op **niveau** van een **kind**.
- Geef elke actie een **directe reactie**.

Verder zijn in dit hoofdstuk enkele visualisaties opgenomen, zoals het gebarenalfabet uitgevoerd in de style uit de styleguide hiernaast, en screenshots van een mogelijk front-end in deze style hieronder. Want, een afbeelding zegt meer dan duizend gebaren.

Voor verdere details voor de vormgeving van het Front-End en de research die hierin is gestoken, verwijzen we graag naar **Bijlage 2 – Analyse & Design Front-End**.



Tijdens het maken van de demo applicatie kwamen we erachter dat er toch behoorlijk wat beperkingen zijn die het doorgeven van de juiste letters en cijfers vermoeilijkt. Buiten het verbeteren van de techniek hebben we ook nagedacht over oplossingen die de uiteindelijke applicatie minder fout gevoelig maakt.

Bij het uitbeelden van een letter of cijfer is er altijd de kans dat het systeem het gebaar als de verkeerde letter of cijfer interpreteert. Dit gebeurt voornamelijk in deze situaties:

- Tijdens de transitie van het ene gebaar naar het andere.
- Als twee gebaren op elkaar lijken.
- Als twee gebaren gelijk zijn, maar dan met een andere beweging.

Letkend op de beperkingen kunnen er keuzes gemaakt worden bij het ontwerpen van een spel of bij het implementeren van de backend functionaliteit. Deze oplossingen kunnen hiervoor ingezet worden:

- Sprint overleg met stakeholder consequenter inzetten
- Nadoen van het getoonde gebaar (Spelkeuze)
- Vertalen van getoonde letter of cijfer naar gebaar (Spelkeuze)
- Een gebaar moet langere tijd getoond worden (Backend)
- Een tweede hand die als enter gebruikt kan worden (Backend)

8 VOORTZETTING VAN HET PROJECT

Bijdrage door : Wally van de Pas 16-12-22 Wally van de Pas 30-12-22

Review door : Ruben Ulfman 16-12-22

De vrijheid van dit project heeft ons dit semester zowel voor- als nadelen opgeleverd. De onderzoeksvraag vanuit de stakeholder hebben wij goed kunnen uitwerken. Echter was het moeilijk om de onderzoeks-gerichte opdracht vanuit Prowise te combineren met de leerdoelen die door onze opleiding geëist werden. Ook zien wij in dat als er verder gegaan zou worden met dit project er een duidelijkere opdracht vanuit Prowise (de stakeholder) zal moeten komen. Dit is dan ook ons advies voor een vervolg van dit project.

De voornaamste punten die zouden moeten verbeteren zijn:

- Sprint overleg met stakeholder consequenter inzetten
- Een gekozen toepassing (door de stakeholder)

Als dit mogelijk is om te organiseren, dan is het voor ons ook mogelijk om voldoende KPI's te kunnen aantonen en kunnen wij niet wachten om vanuit dit onderzoek te gaan starten met de development van een daadwerkelijke applicatie.

BRONNEN EN LITERATUUR

Hieronder is een gealfabetiseerde lijst opgenomen van alle geciteerde bronnen in dit document.

- Brunton, S. (2019, 5 juni). *Intro to Data Science: Overview*. YouTube. <https://www.youtube.com/watch?v=pzo13OPXZS4&feature=youtu.be>
- Centraal Bureau voor de Statistiek. (2022, 30 september). *Jongeren*. Geraadpleegd op 11 november 2022, van <https://www.cbs.nl/nl-nl/visualisaties/dashboard-bevolking/leeftijd/jongeren>
- *Find Open Datasets and Machine Learning Projects* | Kaggle. (z.d.). <https://www.kaggle.com/datasets>
- GOOGLE LLC. *Mediapipe Hands*
Github.io: <https://google.github.io/mediapipe/solutions/hands.html>
- *Hands*. (z.d.). mediapipe. <https://google.github.io/mediapipe/solutions/hands.html>
- <https://datascience.eu/nl/machine-learning/inzicht-in-lstm-netwerken/>
- Lammertink, I. (2019, 21 mei). *Iedereen heeft baat bij kennis van Nederlandse Gebarentaal*. Wetenschap.nu. Geraadpleegd op 21 oktober 2022, van <https://wetenschap.nu/iedereen-heeft-baat-bij-kennis-van-nederlandse-gebarentaal/>
- *Machine learning education* |. (z.d.). TensorFlow. <https://www.tensorflow.org/resources/learn-ml>
- *Nederlands Gebarentaal*. (z.d.). Geraadpleegd op 21 oktober 2022, van <https://www.gebarentaal.nl/vragen+over+NGT>
- *Nederlands Gebarentaal*. (z.d.). <https://www.gebarentaal.nl/vragen+over+NGT>
- NOS. (2019, 16 oktober). *Doven en slechthorenden minder gezond door gebrekkige medische communicatie*. NOS.nl. Geraadpleegd op 21 oktober 2022, van <https://nos.nl/artikel/2306256-doven-en-slechthorenden-minder-gezond-door-gebrekkige-medische-communicatie>
- *OpenCV open-source computer vision*. (z.d.). Geraadpleegd op 14 oktober 2022, van <https://docs.opencv.org/4.6.0/d1/dfb/intro.html>
- Prasad, P. *FingerposeEstimate python module*
Github: <https://github.com/Prasad9/Classify-HandGesturePose/tree/master/pose/utls>
- Renotte, N. (2021, 18 juni). *Sign Language Detection using ACTION RECOGNITION with Python | LSTM Deep Learning Model*. YouTube. <https://www.youtube.com/watch?v=doDUihpi6ro&feature=youtu.be>
- *Tips van een ervaren leraar uit de praktijk*. (z.d.). Geraadpleegd op 3 december 2022, van <https://wij-leren.nl/hoe-kinderen-leren.php>
- Schallwig, A. (2022, 15 november). *Rock, Paper, Scissors with hand gesture recognition*
Medium: <https://medium.com/geekculture/rock-paper-scissors-with-hand-gesture-recognition-841297a7d915>

BIJLAGEN

- 1 R. Ulfman. (2022, 3 december). *(Bijlage) Research Gebarentaal*
- 2 R. Ulfman. (2022, 3 december). *(Bijlage) Analyse & Design Front-End*
- 3 M.v. Driesten. (2022, 15 november). *(Bijlage) UX UI Design*
- 4 M.v. Driesten. (2022, 15 november). *(Bijlage) OpenCV-Python*