OPTIMALISATIE VAN SPRAAKHERKENNINGSTECHNOLOGIE IN ZORGSOFTWARE:

EEN OPLOSSING OM RAPPORTEREN MAKKELIJKER TE MAKEN VOOR ZORGPROFESSIONALS

door

Thomas Haas

Bachelorscriptie
HBO-ICT
Software Engineering
Hogeschool Saxion

Onder begeleiding van Danny Plass
Januari 2023

Bachelorscriptie

Afstudeerder	Thomas Haas
Studentnummer	466290
School	Hogeschool Saxion
Opleiding	HBO-ICT (Software Engineering)
Afstudeerperiode	5 september 2022 – 3 februari 2023
Inleverdatum	22 januari 2023
Afstudeerdocent	Danny Plass
Bedrijf	Avinty
Bedrijfsadres	Enschedesestraat 4, Oldenzaal
Bedrijfsbegeleider	Mark Mooibroek

Voorwoord

Beste lezer,

Voor u ligt de bachelorscriptie “Optimalisatie van spraakherkenningstechnologie in zorgsoftware: een oplossing om rapporteren makkelijker te maken voor zorgprofessionals”. Deze scriptie is geschreven om te voldoen aan de afstudeereisen van de opleiding HBO-ICT (Software Engineering) aan de Hogeschool Saxion in Enschede. Ik ben van september 2022 tot januari 2023 bezig geweest met het onderzoeken en schrijven van mijn scriptie.

Waar ik in projecten zelf altijd de meeste voldoening uithaal is het makkelijker en sneller maken van de workflow voor gebruikers. Daarnaast maakt het leren en toepassen van nieuwe technieken en het ontwikkelen van mobiele apps mij ook enthousiast. Tijdens het afstuderen heb ik gewerkt met technieken waarmee ik geen ervaring had waaronder het trainen en toepassen van Machine Learning modellen. Ik heb ook meer ervaring opgedaan met real time streaming technieken. Daarom heeft deze scriptie mij op zowel persoonlijk als professioneel vlak waardevolle lessen geleerd.

Met dit voorwoord wil ik graag mijn dank uitspreken aan iedereen die heeft bijgedragen aan de afstudeeropdracht. In het bijzonder wil ik mijn bedrijfsbegeleider Mark Mooibroek bedanken voor de uitstekende begeleiding en ondersteuning. Daarnaast wil ik Roy van der Veen bedanken voor zijn input en kennis over Machine Learning. Ook wil ik mijn collega’s bedanken voor hun bijdrage aan de dataset. Daarnaast wil ik de directie, Coen Heck en Niels Kooiker bedanken voor hun rol als product owner. Tot slot wil ik mijn afstudeerdocent Danny Plass bedanken voor de begeleiding, tips en feedback tijdens het schrijven van deze scriptie.

Ik hoop dat deze scriptie een bijdrage kan leveren aan de ontwikkeling van effectievere en gebruiksvriendelijkere zorgsoftware, en dat het kan bijdragen aan een verbeterde zorg voor patiënten.

Ik wens u veel leesplezier toe.

Thomas Haas

Oldenzaal, 22 januari 2023

Samenvatting

Zorgprofessionals zijn ‘s avonds vaak een uur bezig het invoeren van rapportages na overdag meerdere meerdere afspraken met cliënten te hebben gehad. Om dit proces te versnellen wordt de ingebouwde spraakherkenning via het toetsenbord gebruikt. Hierbij is een grote zorg over de veiligheid van het gebruik van 3rd party spraakherkenning applicaties. Daarnaast werkt deze spraakherkenning niet naar wens doordat veel gebruikte afkortingen, medische termen en jargon helemaal niet of niet goed herkend wordt. Daarom wil Avinty een spraak-naar-tekst oplossing die het invoeren van deze rapportages sneller en makkelijker maakt waardoor tussen afspraken door al (concept) rapporten gemaakt kunnen worden.

Het doel van dit onderzoek is om spraakherkenningstechnologie te optimaliseren zodat medische termen herkend kunnen worden. Hiervoor is de volgende onderzoeksvraag opgesteld: Hoe kan spraakherkenningstechnologie geoptimaliseerd worden in zorgsoftware zodat het rapporteren makkelijker gemaakt kan worden voor zorgprofessionals?

Om een antwoord te kunnen geven op de onderzoeksvraag is naast het uitvoeren van een concurrentieanalyse en en het uitzetten van een enquête voornamelijk deskresearch uitgevoerd met betrekking tot de keuze voor de software infrastructuur en spraakherkenningssystemen, waarbij kenmerken zoals performance, mogelijkheid voor real time audio verwerking en optimalisatiemogelijkheden een rol spelen.

Uit de resultaten van deze onderzoeken is gebleken dat een spraakherkenning model die getraind is voor medische doeleinden gemiddeld een tijdsbesparing van 45% kan opleveren. Deze oplossing zorgt voor een toename in werkplezier, minder stress, completere dossiers en verbeterde zorg voor cliënten doordat zorgprofessionals kort na het contactmoment informatie in kunnen spreken. Dit draagt direct bij aan de kwaliteit van zorg.

Op basis hiervan wordt aanbevolen om het prototype en daarbij het spraakherkenning model door te ontwikkelen zodat deze beter presteert op de diversiteit aan spraak die het model zal tegenkomen in de praktijk. Eventueel vervolgonderzoek zou zich kunnen richten op de gebruiksvriendelijkheid van de applicatie of het omgaan met achtergrondgeluid.

Inhoudsopgave

1. Inleiding

Het afstudeerbedrijf Avinty is een softwarebedrijf gevestigd in Oldenzaal. Avinty helpt zorgprofessionals de zorg te verbeteren door software oplossingen te leveren die precies aansluiten bij het vakgebied. Door middel van het Avinty ecosysteem werken deze zorgoplossingen geïntegreerd samen. Hierdoor ontstaat er een optimale gebruikersbeleving, workflow en procesondersteuning voor de zorgprofessional, patiënt en cliënt (Avinty, z.d.).

Twee van deze zorgoplossingen zijn de CARE4 en USER applicaties.

CARE4 is een Elektronisch Cliënten Dossier (ECD) die het administreren makkelijker maakt voor zorgprofessionals. In CARE4 worden de gegevens die de zorg voor de cliënt ondersteunen vastgelegd. Het dossier staat in verbinding met alle betrokkenen, zowel de formele als informele zorg.

USER is een Elektronisch Patiënten Dossier (EPD) die een overzicht over de rapportages van alle disciplines die deelnemen aan het zorgproces van de cliënt weergeeft. Dagelijks vertrouwen meer dan 27.000 zorgprofessionals in de GGZ op USER.

Tijdens het afstuderen zal een software oplossing uitgewerkt worden die het rapporteren makkelijker kan maken voor zorgprofessionals zodat zij ‘s avonds na werktijd niet nog een uur bezig hoeven met het invoeren van rapportages. Deze oplossing zal bestaan uit een softwareapplicatie welke gebaseerd is op verschillende onderzoeken. Het doel van deze onderzoeken is het beantwoorden van de opgestelde deelvragen welke gezamenlijk een antwoord zullen vormen op de hoofdvraag.

Dit document beschrijft hoe de software oplossing tot stand is gekomen, welke keuzes er gemaakt zijn en de verantwoording hiervan. Daarnaast wordt beschreven wat er na mijn afstudeerperiode gedaan kan worden om deze applicatie door te ontwikkelen en te integreren met de huidige software.

1.1 Huidige situatie

De huidige situatie is dat zorgprofessionals op een dag vaak meerdere afspraken met verschillende cliënten hebben. Deze afspraken moeten gerapporteerd worden in de apps van Avinty zoals CARE4 of USER (Avinty, z.d.). Helaas is het zo dat er vrijwel geen tijd is tussen afspraken in om een rapport volledig uit te werken waardoor zorgprofessionals vaak ’s avonds nog een uur bezig zijn met het invoeren van deze rapportages. Daarom wil Avinty een Speech to Text (STT) oplossing die het invoeren van deze rapportages sneller en makkelijker maakt waardoor tussen afspraken door al (concept) rapporten gemaakt kunnen worden.

Momenteel hebben de gebruikers van CARE4 en USER de mogelijkheid om gebruik te maken van de ingebouwde spraakherkenning via het toetsenbord van de smartphone (zie foto). Dit gaat helaas gepaard met een aantal problemen. Op Android is het zo dat het configureren van Speech to Text per device anders kan zijn waardoor zorginstanties verplicht zijn om andere apps te moeten installeren op de devices van zorgprofessionals. Apps zoals de Google Assistent (die levert de Google Voice keyboard) of daarnaast bij Samsung devices vastzitten aan de spraakherkenning van Samsung. Deze spraakherkenning werkt helaas niet naar wens van de zorgprofessionals doordat veel gebruikte afkortingen, medische termen en jargon helemaal niet of niet goed herkend wordt.

Voorbeelden hiervan zijn als volgt: afkortingen waaronder ‘ROM’’ wat staat voor Routine Outcome Monitoring, maar ook namen van bedrijven zoals Mediant en Cimot en namen van afdelingen waaronder bijvoorbeeld afdeling BT en afdeling MZ worden niet herkend door de huidige ingebouwde spraakherkenning.

Daarnaast is er een grote zorg over de veiligheid van het gebruik van 3rd party spraakherkenning applicaties. Doordat er gebruik gemaakt moet worden van externe services er is er geen controle over hoe en waar audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten.

1.2 Gewenste situatie

De gewenste situatie is de ontwikkeling van een softwareproduct bestaande uit een spraakherkenning oplossing die aangestuurd wordt door een Machine Learning model die het invoeren van rapporten voor zorgprofessionals makkelijker kan maken. Deze spraakherkenning moet veelgebruikte afkortingen, medische termen en jargon kunnen herkennen die de ingebouwde spraakherkenning van Apple, Samsung en Google niet herkend. Deze softwareoplossing welke geïntegreerd moeten kunnen worden met de huidige software zal moeten bestaan uit een eigen gebruikersinterface waar gebruikers de mogelijkheid hebben om tekstinvoer te regelen via spraak waarbij medische termen herkend kunnen worden. Deze applicatie moet zelf in-house gehost kunnen worden (of bij een hostingpartij waarbij de juiste ISO en NEN certificeringen aanwezig zijn waaronder de ISO 27001 en NEN 7510). Deze norm is speciaal ontwikkeld voor de zorgsituatie en helpt zorgorganisaties passende beveiligingsmaatregelen te nemen. Op deze manier kan controle behouden worden over hoe audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten (Certificering-keuring.nl, z.d.).

Naast het invoeren van tekst via spraak zou ook de mogelijkheid aangeboden kunnen worden om tekst te kunnen vervangen door spraak, hierdoor kunnen gebruikers vloeiend schakelen tussen invoer via stem (spraak) en aanraking (typen). Het vervangen van tekst via spraak zou gedaan kunnen worden door het selecteren van een stuk tekst welke vervolgens vervangen kan worden door te spreken. Doordat de invoer zowel via spraak en aanraking mogelijk zou kunnen zijn, hebben gebruikers de mogelijkheid om bijvoorbeeld een rapport af te maken door tekst te typen (dit kan handig zijn in het geval dat het zelf getrainde model een bepaald woord toch nog niet herkend heeft).

In het geval dat het model een woord toch nog verkeerd heeft herkend, zou de mogelijkheid er kunnen zijn om een woord snel te vervangen vanuit een lijst met suggesties gebaseerd op de spraakinvoer. Deze suggesties moeten ook vanaf de server komen zonder dat derden toegang krijgen tot deze data.

Ter verduidelijking, de lijst met suggesties zou bijvoorbeeld kunnen verschijnen in de vorm van een kleine pop-up zodra er op een woord geklikt wordt. Zodra een woord vervangen is door een suggestie kan deze feedback teruggestuurd worden naar het model. Het model kan op deze manier leren van de feedback en zichzelf blijven trainen. Natuurlijk behoudt de gebruiker ook de mogelijkheid om de tekst te vervangen door middel van het gewenste resultaat te typen.

Hierbij is voor Avinty het belangrijkste doel de realisatie van een prototype welke geïntegreerd kan worden met de huidige software waarmee aangetoond kan worden dat een Machine Learning model woorden kan herkennen welke het standaard model niet of niet goed herkend na het trainen van het model op een eigen dataset.

2. Aanpak

In het Plan van Aanpak is gekozen om het project uit te voeren aan de hand van de productontwikkelingscyclus, deze cyclus wordt gebruikt als overkoepelend framework welke de volgorde bepaald van het softwareontwikkelingsproces.

2.1 Productontwikkelingscyclus

De productontwikkelingscyclus (Byrne, z.d.) legt de verschillende fasen uit die nodig zijn om een product te ontwikkelen. Het model geeft een duidelijk beeld van de stappen die gezet moeten worden om tot een eindproduct te komen dat aansluit bij de doelgroep.

1. Concept
In de conceptfase worden de behoeftes van gebruikers in kaart gebracht door middel van een probleem- en behoefteanalyse. De resultaten uit deze fase zullen een basis vormen voor de ontwerp en ontwikkelingsfase.

2. Haalbaarheid
In de haalbaarheidsfase wordt de haalbaarheid onderzocht door onderzoek te doen naar complexe onderdelen en uitdagingen die tijdens de realisatie kunnen voorkomen.

3. Ontwerp en ontwikkeling
In de ontwerp en ontwikkeling fase worden de product requirements opgesteld. Voorafgaand aan de ontwikkeling zullen eerst de deelvragen in deze fase beantwoord moeten om goed onderbouwde keuzes te kunnen maken. Vervolgens kan de software oplossing ontwikkeld worden gebaseerd op de vereisten.

4. Testen
In de testfase wordt het eindproduct getest om er zeker van te zijn dat het product voldoet aan de eisen die opgesteld zijn in de eerdere fases, en om zeker te zijn dat er geen fouten zijn gemaakt tijdens de realisatie. Daarnaast zal de nauwkeurigheid van het model worden getest en gekeken worden of er punten zijn die verbeterd moeten worden.

5. Validatie
In de validatiefase wordt er gekeken of de gebruikers problemen opgelost zijn. Als dat zo is, is het product klaar om de geintegreerd te worden met de software van Avinty.

---

6. Integratie* **(Deze fase is hernoemd zodat het beter bij de opdracht past)**
In de integratiefase (officieel lanceringsfase) wordt de applicatie geïntegreerd met de bestaande software. Vervolgens zal er een nieuwe versie van de Avinty software beschikbaar gemaakt worden. Deze fase is hernoemd omdat er eerst een integratie nodig is met de huidige software voordat het product gelanceerd kan worden.

7. Verbeteren*
Nadat het product is geïntegreerd, wordt de feedback geanalyseerd welke gebruikt wordt om verbeteringen te adviseren. Doordat de integratie buiten de scope van het project valt wordt in deze fase onderzocht welke strategieën gebruikt kunnen worden om de nauwkeurigheid van de spraakherkenning applicatie te verhogen na implementatie.

2.2 Onderzoeksvragen

Om het afstuderen richting te geven en tot een goed einde te brengen, is een hoofd- en zijn deelvragen geformuleerd. De resultaten van deze deelvragen geven samen antwoord op de hoofdvraag. De deelvragen zijn ondergebracht in verschillende fases gebaseerd op de productontwikkelingscyclus.

Hoofdvraag

Hoe kan spraakherkenningstechnologie geoptimaliseerd worden in zorgsoftware zodat het rapporteren makkelijker gemaakt kan worden voor zorgprofessionals?

Deelvragen

Concept

- Wie zijn de belanghebbenden?
- Wat zijn de behoeften van de belanghebbenden?
- Welke features bieden concurrenten aan?
- Welke spraakherkenningstechnologieën zijn momenteel beschikbaar en welke van deze technologieën zijn geschikt voor gebruik in zorgsoftware?

Haalbaarheid

- Wat is de haalbaarheid van het concept?
- Welke middelen zijn nodig om de spraakherkenningstechnologie te optimaliseren?

Ontwerp en ontwikkeling

- Welke functionaliteiten moet de applicatie krijgen?
- Wat zijn de voor- en nadelen van dataverwerking in de front- vs. de back-end?
- Welke onderdelen komen in de front-, en welke komen in de back-end?
- Hoe kan een model zo efficiënt mogelijk getraind worden?
- Hoe kan een dataset voor het trainen het beste samengesteld worden?

Testen

- Wat is de nauwkeurigheid van het model na training?
- Voldoet het product aan alle vereisten?
- Zijn er punten die verbeterd moeten worden?

Validatie

- Is spraakinvoer via het model sneller en makkelijker dan typen?

Integratie*

- Hoe kan het product geïntegreerd worden met de huidige software?
- Welke specifieke uitdagingen komen er kijken bij het uitrollen van spraakherkenningstechnologie in zorgsoftware?

Verbeteren*

- Welke features kunnen worden ingezet om het prototype en daarbij de gebruikerservaring te verbeteren na het uitrollen?

* De integratie- en de verbeteringfase vallen buiten de scope van het project. Overigens zullen deze deelvragen beantwoord worden door middel van een advies en aanbevelingen.

2.3 Onderzoeksmethoden

De productontwikkelingscyclus geeft een globale beschrijving hoe de verschillende fases eruit zien. Deze paragraaf gaat daar verder op in door een specifiek beeld te geven per fase over welke onderzoeksmethoden gebruikt worden en welke overige werkzaamheden er uitgevoerd worden. De onderzoeksmethoden zijn gebaseerd op het DOT framework (Heck, 2020) en zijn ondergebracht in verschillende fases gebaseerd op de productontwikkelingscyclus.

Concept

In de conceptfase van de productontwikkelingscyclus zal een onderzoek onder de gebruikers van CARE4 en USER uitgevoerd worden. Dit onderzoek is bedoeld om pijnpunten te achterhalen die zorgprofessionals momenteel ondervinden bij het rapporteren in de huidige software. Dit wordt gedaan door met verschillende onderzoeksmethoden te onderzoeken welke problemen deze doelgroep heeft. Het gebruikersonderzoek zal plaatsvinden door middel van een enquête.

Daarnaast zal door middel van een concurrentieonderzoek onderzocht worden welke functionaliteiten concurrenten aanbieden en of deze functies interessant zijn om mee te nemen in deze applicatie. De resultaten van deze onderzoeken zullen de basis vormen voor de ontwerp en ontwikkelingsfase.

Haalbaarheid

In de haalbaarheidsfase wordt de haalbaarheid gevalideerd, dit moet mogelijke problemen voorkomen tijdens de realisatie. Het valideren van de haalbaarheid zal bestaan uit het onderzoeken van complexe onderdelen en uitdagingen die tijdens de realisatie kunnen voorkomen. Dit haalbaarheidsonderzoek bevat onderzoek hoe spraak naar tekst met spraakherkenning werkt en hoe de optimalisatie (fine-tuning) gerealiseerd kan worden zodat het model veelgebruikte afkortingen, medische termen en jargon kan herkennen.

Ontwerp en ontwikkeling

In de ontwerp en ontwikkelingsfase worden de functionele requirements eisen opgesteld waaraan het product moet voldoen. Voorafgaand de productrealisatie zal theoretisch onderzoek uitgevoerd worden die de deelvragen in de ‘ontwerp en ontwikkeling’ fase zullen beantwoorden. Het beantwoorden van deze deelvragen door middel van onderzoek is nodig om logische keuzes te kunnen maken voor de realisatie.

Zodra het theoretische onderzoek voltooid is kan de productrealisatie beginnen waarbij de softwareontwikkeling richtlijnen van Avinty nagestreefd moeten worden. Voordat een taak als voltooid bestempeld mag worden ("Definition of Done"), moet de code gezien zijn door minimaal 2 engineers (de ontwikkelaar + minimaal één andere collega). Deze methodiek zorgt ervoor dat de code op kwaliteit blijft. Tijdens de ontwikkeling wordt blijvend gecontroleerd door middel van testen of aan de vereisten wordt voldaan.

De productrealisatie bestaat uit het realiseren van een proof of concept. Het proof of concept zal bestaan uit een applicatie waar de meest uitdagende onderdelen in zitten waaronder: het streamen van audio naar de server, het verwerken van deze audio op de server, het terugkrijgen van tekst en daarnaast het trainen van een model zodat medische termen herkend kunnen worden in de applicatie.

Testen
In de testfase wordt het software systeem uitgebreid getest om er zeker van te zijn dat er geen fouten zijn gemaakt tijdens de ontwikkeling en dat het product voldoet aan de eisen die opgesteld zijn in de eerste fases. Daarnaast zal de nauwkeurigheid van het model worden getest en gekeken worden of er punten zijn die verbeterd moeten worden.

Het bepalen van de nauwkeurigheid gebeurt door het berekenen van de Word Error Rate (WER) van het model. De Word Error Rate is een veelgebruikte indicator van nauwkeurigheid van spraak-naar-tekst oplossingen. De Word Error Rate geeft de foutmarge aan ten opzichte van een correct transcript.

Validatie
In de validatiefase wordt er getest of de applicatie het probleem daadwerkelijk opgelost heeft. Dit gebeurt door de tijd te meten van dezelfde invoer via spraak en typen door iemand die een bepaalde tekst nog niet eerder heeft gezien. Als de invoer via spraak sneller en/of makkelijker is dan typen, is de applicatie klaar om geïntegreerd te worden met de huidige software.

Integratie*
In de integratiefase is het tijd om het product te introduceren aan de gebruikers van CARE4 en USER. Dit bestaat uit het integreren van het prototype met de Avinty applicaties. Doordat deze fase buiten de scope van het project valt zal in deze fase een advies opgesteld worden hoe het softwaresysteem het beste geïntegreerd kan worden met de huidige software. Daarnaast wordt onderzocht welke uitdagingen er komen kijken bij het implementeren van het prototype in de huidige software.

Verbeteren*
In de verbeteringsfase wordt onderzocht welke features ingezet kunnen worden om het prototype en daarbij de gebruikerservaring te verbeteren na het uitrollen. Doordat deze fase buiten de scope van het project valt zal in deze fase een advies opgesteld worden over hoe het prototype verbeterd kan worden.

* De integratie- en de verbeteringfase vallen buiten de scope van het project. Overigens zullen deze deelvragen beantwoord worden door middel van een advies en aanbevelingen.

3. Fase 1: Concept

In de conceptfase van de productontwikkelingscyclus wordt onderzocht welke features de doelgroep nodig heeft. Dit wordt gedaan door het onderzoeken welke features concurrenten aanbieden om te kijken of deze features kunnen helpen met het oplossen van het gebruikersprobleem. Daarnaast wordt een enquête uitgezet die onderzoekt welke problemen en wensen de doelgroep heeft, maar ook een inzage geeft in het gebruikersgedrag van de doelgroep. De resultaten van deze onderzoeken zullen de basis vormen voor de functionele- en niet functionele requirements in de ontwerp en ontwikkelingsfase.

3.1 Concurrentieanalyse

Bij de concurrentieanalyse wordt er gekeken naar bedrijven en mobiele apps die vergelijkbare oplossingen aanbieden die vallen onder minimaal één van de volgende voorwaarden:

- Mobiele apps die algemene spraak-naar-tekst services aanbieden;
- Bedrijven die spraak oplossingen aanbieden specifiek voor de zorg;
- Bedrijven die spraak oplossingen aanbieden voor overige specifieke doeleinden.

3.1.1 Concurrenten

Hieronder volgen de onderzochte concurrenten, vervolgens worden deze concurrenten vergeleken in een tabel en ten slotte wordt deze informatie samengevat in een conclusie.

Attendi (Attendi Speech Service)

Voordelen & Features

Het spraak-naar-tekst model is specifiek getraind voor GGZ doeleinden.

De technologie kan omgaan met een verschillende omgevingsgeluiden. De gebruiker krijgt een terugkoppeling als er te veel omgevingsgeluid is.

De ingesproken tekst kan als concept opgeslagen worden zodat deze later verfijnd kan worden tot een eindresultaat. Want onderweg of bij een cliënt is niet het moment om een ingesproken tekst te controleren en te verwerken tot een eindresultaat. Nieuwe ingesproken stukken tekst die nog niet aangepast zijn worden gemarkeerd zodat het duidelijk is dat dit een concept bevat.

Het is het mogelijk om de spraakinvoer te pauzeren en te hervatten. Tijdens het inspreken kan de zorgprofessional onderbroken worden of je wil als begeleider heel even nadenken wat je gaat inspreken.

Nieuwe binnenkomende audio data wordt gebruikt om het model te verbeteren. De audioberichten en bijbehorende transcripten worden gebruikt als trainingsmateriaal voor de spraak-naar-tekst-modellen. Hierdoor wordt de nauwkeurigheid van het model constant verbeterd.

Er zijn regels geïmplementeerd waardoor woorden omgezet kunnen worden naar een vorm die gebruikelijk is in de rapportages. “Tweehonderd vijftig milliliter” wordt dan omgezet naar “250 ml” en “De heer” wordt “Dhr.”.

Er wordt geïnformeerd over de risico's van de technologie en hoe daar goed mee om te gaan. Opnemen in een rustige omgeving is niet alleen beter voor de kwaliteit van spraak naar-tekst, maar ook een stuk veiliger.

ISO 27001 en NEN 7510 certificering is aanwezig, dit geeft aan dat de organisatie vertrouwelijk en integer omgaat met de patiëntgegevens (Certificering-keuring.nl, z.d.).

Nadelen
De service is behoorlijk prijzig, de service van Attendi kost namelijk €3,50 per maand per gebruiker (voor alleen de spraak naar tekst functionaliteit).

Nuance (Dragon Medical One)

Voordelen & Features
Het spraak-naar-tekst model is specifiek getraind voor GGZ doeleinden.

Iedere zorgprofessional heeft een spraakprofiel met de standaard medische woorden en eigen jargon. Tijdens het eerste gebruik wordt er een spraakprofiel gemaakt. Door fouten te corrigeren went Dragon aan je stem. Deze nieuwe techniek zorgt ervoor dat de herkenning na 10 minuten belachelijk hoog is.

ISO 27001 en NEN 7510 certificering is aanwezig, dit geeft aan dat de organisatie vertrouwelijk en integer omgaat met de patiëntgegevens (Certificering-keuring.nl, z.d.).

Nadelen
De service is extreem prijzig, de service van Nuance kost namelijk €71,- per maand per gebruiker. Daarnaast wordt service ondersteuning aangeboden tegen een bedrag van €15,- per maand per gebruiker.

Amazon Transcribe Medical

Voordelen & Features
Amazon Transcribe Medical is een service voor automatische spraakherkenning. Gesprekken tussen zorgverleners en patiënten vormen de basis voor het diagnose- en behandelplan van een patiënt en de workflow voor klinische documentatie. Aangedreven door state-of-the-art machine learning, Amazon Transcribe Medical kan medische terminologie zoals namen van medicijnen, procedures en zelfs aandoeningen of ziekten nauwkeurig transcriberen. Daarnaast biedt Amazon ook de mogelijkheid om een samenvatting van een gesprek te genereren.

Nadelen
De prijs is erg variabel, de eerste 250,000 minuten kosten $0.00600 per minuut. Daarna zijn er verschillende tiers afhankelijk van hoeveel minuten er gebruikt worden. Bij het gebruik van services van Amazon is er geen controle over hoe audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten. Amazon heeft geen certificaten specifiek voor het vertrouwelijk en integer omgaan met de patiëntgegevens.

Apple, Google en Samsung

Voordelen & Features

Het grootste voordeel van de spraak-naar-tekst services van Apple, Google en Samsung is dat deze gratis te gebruiken zijn. Daarnaast zijn deze services ingebouwd in het toetsenbord wat zorgt voor een makkelijke switch tussen spraak- en tekstinvoer.

Nadelen

Bij het gebruik van services van Apple, Google en Samsung is er geen controle over hoe audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten. Apple en Samsung hebben geen certificaten specifiek voor het vertrouwelijk en integer omgaan met de patiëntgegevens. Google Cloud beschikt overigens wel de bijbehorende internationale certificaten en houdt zich daardoor aan de internationale normen.

3.1.2 Vergelijking van concurrenten

In de onderstaande tabel worden de sterke en zwakke punten van de concurrenten weergegeven.

Het doel van de sterke punten is het meenemen van deze ideeën wanneer deze bij de opdracht passen en kunnen helpen met het oplossen van het gebruikersprobleem. De zwakke punten en de conclusie hieronder beschrijven waarom het niet logisch is om voor een concurrent te kiezen in plaats van het ontwikkelen van een eigen model.

<table>
<tr><th>Sterke punten</th><th>Attendi</th><th>Nuance Dragon Medical One</th><th>Amazon Transcribe Medical</th><th>Apple, Google en Samsung</th></tr>
<tr><td rowspan="10"></td><td>Omgaan met omgevingsgeluiden</td><td rowspan="2">Elke zorgprofessional heeft een spraakprofiel</td><td rowspan="4">Groot bereik van medische termen die herkend worden (namen van medicijnen, procedures, etc…)</td><td rowspan="4">Makkelijk te switchen tussen spraak en tekst</td></tr>
<tr><td>Opslaan als concept</td></tr>
<tr><td>Pauzeren en hervatten van invoer</td><td rowspan="4">Door fouten te corrigeren went Dragon Medical One aan jouw stem</td></tr>
<tr><td>Afkortingen vervangen</td></tr>
<tr><td>Model doortrainen door audio en transcripten</td><td rowspan="2">Mogelijkheid om een samenvatting van een gesprek te genereren</td><td>Standaard ingebouwd in het toetsenbord</td></tr>
<tr><td>Data wordt gehost in Nederland</td><td>Google Cloud: ISO 27001 en NEN 7510 certificering</td></tr>
<tr><td colspan="2">ISO 27001 en NEN 7510 certificering</td><td colspan="2">Veel kapitaal om een hogere nauwkeurigheid te bereiken</td></tr>
<tr><td colspan="3">Specifiek getraind voor GGZ en/of medische doeleinden</td><td>Gratis te gebruiken</td></tr>
<tr><td rowspan="3">Zwakke punten</td><td colspan="3">Zeer prijzig</td><td></td></tr>
<tr><td colspan="4">Geen controle over hoe audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten</td></tr>
<tr><td></td><td colspan="3">De data wordt gehost in het buitenland</td></tr>
</table>

3.1.3 Conclusie concurrentieanalyse

De meest geschikte bedrijven die een bestaande oplossing bieden voor het gebruikersprobleem zijn Nuance en Attendi. Bij deze keuze speelt de aanwezigheid van een specifiek getraind model voor medische doeleinden en de aanwezigheid van de ISO 27001 en NEN 7510 certificeringen een belangerijke rol. Deze certificeringen geven aan dat de organisatie vertrouwelijk en integer omgaat met de patiëntgegevens (Certificering-keuring.nl, z.d.).

Het grootste nadeel is helaas dat bedrijven die een specifiek getraind model aanbieden voor medische doeleinden en de juiste certificeringen hebben enorm hoge prijzen vragen voor hun services. De maandelijkse kosten voor Nuance bij 27.000 gebruikers (USER) zou uitkomen op €1.9M per maand (€71,- per maand per gebruiker). In het geval van Attendi is dit ongeveer €100K per maand (€3,50 per maand per gebruiker).

Bij Amazon, Apple en Samsung zijn geen ISO 27001 en NEN 7510 certificeringen aanwezig. Daarom is er niet genoeg controle over hoe audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten. Google Cloud leeft bovenstaande internationale normen wel na (Google Cloud, z.d.), maar biedt helaas geen model aan die specifiek is getraind voor medische doeleinden.

Attendi biedt overigens wel veel functionaliteiten aan die voor de applicatie van Avinty erg interessant zijn waaronder het omgaan met omgevingsgeluiden, het pauzeren en hervatten van invoer, het vervangen van afkortingen en het doortrainen van het spraak-naar-tekst model op basis van invoer bestaande uit audio en transcripten.

De Dragon Medical One van Nuance heeft een interessante functionaliteit waarbij elke zorgprofessional een apart spraakprofiel heeft met standaard medische woorden en eigen lijst van jargon.

Tot slot zijn de concurrenten die spraak-naar-tekst modellen specifiek getraind hebben voor GGZ en/of medische doeleinden zeer prijzig. Overige bedrijven die algemene spraak-naar-tekst oplossingen aanbieden zijn gratis te gebruiken, maar kunnen geen medische termen herkennen.

Doordat de kosten van services die modellen specifiek voor ggz en/of medische doeleinden aanbieden zo hoog zijn is dit geen geschikte oplossing. Daarom is het ontwikkelen van een eigen model een logische keuze. Op deze manier kan niet alleen controle behouden worden over hoe en waar audio/data verwerkt wordt welke bestaat uit gevoelige informatie van cliënten, maar ook met welke data er getraind wordt.

3.2 Enquête

3.2.1 Inleiding en doel van de enquête

In de voorbereidende fase is de huidige- en gewenste situatie geanalyseerd, maar na deze analyse zijn er nog steeds een aantal vraagstukken over. Daarom wordt in deze fase onderzocht waar softwarematig de pijnpunten en winsten (pains and gains) liggen onder de gebruikers. Daarnaast wordt ook het gebruikersgedrag onderzocht en wordt er gekeken welke suggesties de zorgprofessionals zelf hebben. Zodra deze informatie bekend is, kan er nagedacht worden over specifieke functionaliteiten voor de applicatie.

3.2.2 Theoretisch kader

a) Enquête software

De enquête software die benodigd is om deze enquête op te zetten moet voldoen aan een aantal specificaties om het onderzoek te optimaliseren. De lijst met criteria waaraan de software moet voldoen, gezien mijn wensen, is dat de mogelijkheid er moet zijn om:

- Bepaalde specifieke vragen over te slaan;
- Een vraag te beantwoorden via de Likert-schaal (Joshi, Kale, Chandel, & Pal, 2015, blz. 400);
- Tekstinvoer toe te staan bij een antwoord;
- Een duidelijk rapport van de resultaten te genereren en een optie om de resultaten te downloaden.

Saxion raadt aan om Qualtrics te gebruiken voor het opzetten van een enquête. Qualtrics voldoet ook aan de opgestelde criteria voor de enquête software. Daarom wordt Qualtrics gebruikt om de enquête op te zetten. Qualtrics is een uitgebreid softwareprogramma voor enquêtes die gratis te gebruiken is met mijn studentenmail van Saxion.

b) Stappenplan

Voor de opzet van de enquête wordt gebruik gemaakt van een stappenplan. Dit stappenplan beschrijft hoe de resultaten van het marktonderzoek verwerkt kan worden tot concrete vragen voor de enquête (Hoftijzer & Korter, 2013).

1. Bepaal het doel van het onderzoek;
2. Formuleer het probleem;
3. Stel de subvragen in;
4. Bepaal de indicatoren (meetbare onderwerpen);
5. Stel de enquêtevragen in;
6. Stel de definitieve enquête in.

c) Vragen criteria

Ook is er een lijst met criteria opgesteld waaraan de vragen moeten voldoen naar aanleiding van “Dit is onderzoek!” uit Baarda (2019). Voordat de enquête verspreid zal worden, zal gecontroleerd worden of de vragen aan de volgende criteria voldoen.

- De vragen moeten passen bij het niveau van de respondent;
- De vragen moeten taalkundig duidelijk zijn;
- De vragen moeten concreet zijn (gebruik bijvoorbeeld niet het woord ‘regelmatig’);
- De vragen mogen niet meer dan één vraag bevatten;
- De vragen moeten relevant zijn voor de respondent en kunnen worden beantwoord door de beantwoorder;
- De vragen mogen geen kennis veronderstellen die de respondent misschien niet heeft.

Omdat deze criteria aangehouden worden zullen de vragen correct, duidelijk en gemakkelijk te beantwoorden zijn voor de respondenten (Baarda, 2019).

3.2.3 Verantwoording van de enquête

Deze enquête zal een schriftelijke online gedistribueerde enquête zijn. Dit onderzoek zal gestructureerd en individueel zijn. Volgens Baarda (2019) zijn de redenen om te kiezen voor een gestructureerd en individueel onderzoek:

- Het onderzoek zal snel, direct en duidelijk zijn;
- De distributie zal intern plaatsvinden, intern zijn er lijntjes naar zorgprofessionals/gebruikers;
- De kosten (geld en tijd) zullen relatief laag zijn;
- De tijd van respons verzameling zal relatief laag zijn;
- Anonimiteit is mogelijk.

3.2.4 Opbouw van de enquête

De opbouw van de enquête moet zo duidelijk mogelijk gestructureerd zijn. Er mag geen ruimte zijn voor de deelnemers om iets op een eigen manier te interpreteren, dit om verkeerde interpretaties te voorkomen. De enquête begint met een inleiding over het onderzoek en het onderwerp. Vervolgens worden algemene vragen gesteld zoals het geslacht en de leeftijdscategorie. Deze algemene informatie kan later worden gebruikt om verbanden te vinden.

De enquête zal voornamelijk vragen bevatten die beantwoord kunnen worden door schalen in te vullen. Hierbij wordt de 7-punts Likertschaal gebruikt. De Likertschaal (Joshi, Kale, Chandel, & Pal, 2015, p. 400) is een methode om gegevens te voorzien van een ordinaal meetniveau dat moeilijk te kwantificeren is. De 7-punts Likert-schaal zal in dit geval gebruikt worden in plaats van de 5-punts Likert-schaal omdat dit meer specifieke antwoorden oplevert (Onderzoekdoen.nl, 2017).

Daarnaast wordt het responspercentage en de kwaliteit van de antwoorden gemaximaliseerd. Mensen zullen eerder afhaken als een enquête te lang is. Daarom wordt aangeraden ervoor te zorgen dat een enquête niet langer duurt dan 8 minuten. Dit betekent dat er maximaal 20 tot 25 korte en bondige (open of gesloten) vragen gesteld kunnen worden (Benders, 2022).

3.2.5 Verspreiding van de enquête

De enquête zal intern verspreid worden via een (anonieme) link. Hierbij is aan GGZ pilot klanten gevraagd de enquête in te vullen en aan het accountteam van Jeugdzorg gevraagd de enquête te verspreiden.

Om het minimum aantal respondenten voor de enquête te kunnen berekenen, moet de volgende formule gebruikt worden. Het berekenen van het minimum aantal respondenten is belangrijk omdat het resultaat van deze formule aangeeft hoeveel respondenten er minimaal nodig zijn om een goede conclusie te kunnen trekken (SurveyMonkey, z.d.).

$$\frac{\frac{z^2 \times p(1-p)}{e^2}}{1+(\frac{z^2 \times p(1-p)}{e^2 N})}$$

De formule bevat verschillende variabelen:

Waarde	Omschrijving
N	De populatie; in dit geval het aantal gebruikers van CARE4 (onbekend) en USER (27.000) bij elkaar opgeteld (Avinty, 2022).
z	Betrouwbaarheidsniveau; bij een betrouwbaarheidsniveau van 95% is de z-score 1,96. Standaarden die algemeen worden gebruikt door onderzoekers zijn 90%, 95% en 99%.
e	Foutenmarge; de foutenmarge bij een populatieomvang van 27.000 en een betrouwbaarheidsniveau van 95% is 10%.
p	Percentage waarde; wanneer een enquête voor het eerst uitgevoerd wordt en omdat de meeste enquêtes meer dan één vraag hebben, wordt er aangeraden om p = 0,5 te gebruiken. Dit leidt tot een geschatte steekproefgrootte die niet te krap en niet te ruim is.

Met de genoemde formule en de ingevoerde variabelen aanbevolen door SurveyMonkey is het resultaat dat de enquête 96 respondenten nodig heeft om een goede conclusie te kunnen trekken (SurveyMonkey, z.d.).

Het onbekende gebruikersaantal van CARE4 heeft geen invloed op het aantal benodigde respondenten. Wanneer een hogere waarde ingevuld wordt als N-waarde verandert het minimum aantal respondenten naar 97 respondenten (dat is een verschil van 1).

3.2.6 Enquêtevragen

De enquêtevragen kunnen gevonden worden in Appendix A.

3.2.7 Enquêteresultaten

De enquête heeft 132 responses. Daarmee is het doel van 96 respondenten behaald wat betekent dat er genoeg respondenten zijn om een goede conclusie te kunnen trekken.

De enquêteresultaten kunnen gevonden worden in Appendix B.

3.2.8 Conclusie enquête

Uit zowel de data als de reacties van de respondenten is gebleken dat een spraak-naar-tekst oplossing direct zorgt voor een tijdsbesparing wat resulteert in een afname in werkdruk en meer tijd voor de cliënt.

Respondenten geven aan dat “een spraak-naar-tekst oplossing zeker een hele verbetering in het werkproces zou kunnen geven”. Een andere respondent heeft als opmerking “het invoeren van rapportages valt onder cliënt tijd. Bij een snellere vorm van rapporteren is er meer tijd over voor de cliënt”.

De meeste respondenten maken tussen de verschillende bezoeken op een dag korte aantekeningen op een notitieblok of in een notitie-applicatie op hun smartphone. Deze aantekeningen werken ze dan later uit. Om verschillende redenen is dit niet ideaal:

- Door het later op de dag uitwerken van aantekeningen gaan belangrijke details verloren;

- Wanneer er tussen bezoeken door geen tijd is om belangrijke informatie te noteren, wordt er door de respondenten stress ervaren;
- Via de smartphone een rapportage typen is onhandig, de omvang van de rapportage is te groot om via een klein toetsenbord te verwerken;
- Aan het einde van de dag alle rapportages afmaken is geen prettige afsluiting van de dag.

Als zorgprofessionals onderweg zijn kost het veel tijd om een laptop te pakken, de applicatie op te starten en daarna te gaan typen. In de praktijk is hier vaak geen tijd voor waardoor het lastig is om direct na een contactmoment te rapporteren. Spraak naar tekst werkt altijd en overal en is een laagdrempelige manier om een rapportage te maken. De technologie past goed in het werkproces waardoor het voor de zorgprofessionals veel makkelijker is om dit soort momenten te benutten met het maken van een concept rapportage.

Vergeet nooit meer een belangrijk detail
Na ieder cliënt bezoek een rapportage schrijven blijkt in de praktijk voor de zorgprofessionals niet eenvoudig te zijn. Dit zorgt voor stress omdat veel belangrijke informatie mogelijk verloren gaat. Door kort na het contactmoment de informatie in te spreken gaan belangrijke details nauwelijks verloren. Dit draagt direct bij aan de kwaliteit van zorg.

Minder stress
Uit de resultaten van de enquête blijkt dat de zorgprofessionals minder stress denken te ervaren als ze direct na een bezoek al informatie kunnen inspreken. Doordat ze weten dat alle informatie een plek heeft gekregen sluiten ze het vorige bezoek beter af. Daarnaast sluiten ze ook de dag prettiger af. In plaats van alle rapportages in zijn volledigheid te typen, verfijnen ze nu een basis die eerder op de dag al is ingesproken.

De mogelijkheid om rapportages altijd en overal inspreken met behulp van een spraak-naar-tekst systeem resulteert in:
- Completere dossiers en minder stress voor de zorgprofessionals;
- Een tijdsbesparing waardoor er meer tijd is voor de cliënten;
- Een toename in werkplezier.

4. Fase 2: Haalbaarheid

In de haalbaarheidsfase wordt de haalbaarheid gevalideerd, dit moet mogelijke problemen voorkomen tijdens de realisatie. Deze onderzoeksresultaten zullen de basis vormen voor verdere onderzoeken en conclusies die getrokken worden in de volgende fase.

Door vroegtijdig vraagstukken te onderzoeken zal er tijdens de realisatiefase van het project aanzienlijk minder problemen ondervonden worden. De grootste uitdaging is het begrijpen hoe een spraakherkenningssysteem werkt, uit welke componenten deze bestaat en hoe dit systeem geoptimaliseerd kan worden zodat medische termen herkend worden.

4.1 Hoe werkt een spraakherkenningssysteem?

Een spraakherkenningssysteem bestaat uit verschillende componenten die samenwerken om spraak om te zetten in tekst. De belangrijkste componenten zijn (Medium, 2021):

- **Een spraakopname apparaat:** dit kan een microfoon, een telefoon of een ander apparaat zijn dat audio-opnamen kan maken.

- **Een spraakherkenning algoritme:** dit is de kern van het systeem, dit component is verantwoordelijk voor omzetten van een spraakopname in een transcriptie.

- **Een tekstverwerkingssysteem:** dit is verantwoordelijk voor het verwerken van de transcriptie en het maken van een leesbaar document of bestand.

- **Een gebruikersinterface:** dit is de manier waarop de gebruiker toegang heeft tot het systeem en de transcriptie dan bekijken en eventueel bewerken.

Voor deze opdracht is het spraakherkenning algoritme (ook wel een spraak-naar-tekst model genoemd) van het grootste belang. Doordat dit component verantwoordelijk is voor de conversie van spraak naar tekst ligt daar de focus op. Om ervoor te zorgen dat een model medische termen kan herkennen zal deze getraind moeten worden, dit kan op meerdere manieren. Dit kan door het trainen van een nieuw model vanaf scratch, of door het verder trainen van een model die al een goede basis heeft van de Nederlandse taal.

4.2 Bestaand of nieuw model?

In deze paragraaf wordt het hertrainen van een bestaand model vergeleken met het trainen van een nieuw model vanaf scratch. Er zijn enkele belangrijke voordelen om een bestaand machine learning model te hertrainen in plaats van een nieuw model te trainen (MATLAB & Simulink, z.d.):

- **Snelheid:** het hertrainen van een bestaand model kan sneller zijn dan het trainen van een nieuw model vanaf scratch, omdat het model al een basis heeft en alleen nieuwe gegevens nodig heeft om te leren in plaats van alle gegevens te moeten gebruiken om te leren.

- **Kosten:** het hertrainen van een bestaand model is vaak goedkoper dan het trainen van een nieuw model, omdat het minder tijd en middelen kost om te hertrainen.

- **Betrouwbaarheid:** als een model al is getraind en getest en goede resultaten heeft behaald, is het waarschijnlijk betrouwbaarder dan een nieuw model dat nog niet is getest. Het hertrainen van een bestaand model kan ervoor zorgen dat het model nog betrouwbaarder wordt, omdat het is aangepast aan nieuwe gegevens.

- **Flexibiliteit:** het hertrainen van een bestaand model kan handig zijn als een model aangepast moet worden aan nieuwe gegevens. In plaats van een nieuw model te moeten trainen, kan een bestaand model hertraind worden en aangepast worden aan de behoefte van de gebruikers. Het hertrainen van een model voor een andere taak wordt ook wel Transfer Learning genoemd.

4.3 Transfer Learning

Transfer learning is een Deep Learning-benadering waarbij een model dat voor één taak is getraind, wordt gebruikt als uitgangspunt voor een model dat een vergelijkbare taak uitvoert. Het aanpassen en omscholen van een netwerk met Transfer Learning is meestal veel sneller en gemakkelijker dan een netwerk helemaal opnieuw trainen. Deze aanpak wordt onder meer gebruikt voor objectdetectie, beeldherkenning en spraakherkenning toepassingen.

Transfer Learning is een populaire techniek omdat hiermee modellen getraind kunnen worden door middel van minder gelabelde gegevens door populaire modellen te hergebruiken die al zijn getraind op grote gegevenssets. Hierdoor is er minder trainingstijd en rekenkracht vereist (MATLAB & Simulink, z.d.).

Door middel van Transfer Learning kan een bestaand model bijvoorbeeld Kaldi, DeepSpeech, of Whisper die gebruik maakt van het neurale netwerk van OpenAI getraind kunnen worden.

4.4 Wat is een neuraal netwerk?

Een neuraal netwerk (NN) is een type van kunstmatige intelligentie dat kan worden gebruikt voor verschillende doeleinden, waaronder spraakherkenning. Een spraak-naar-tekst systeem is een specifiek soort systeem dat ontworpen is om spraak om te zetten in tekst. Dit kan worden gedaan met behulp van verschillende technieken, waaronder het gebruik van een neuraal netwerk.

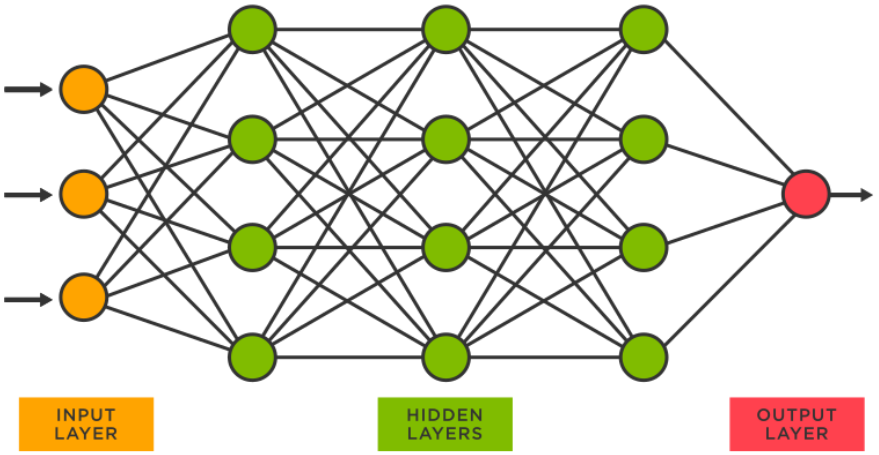
In het algemeen kan een neuraal netwerk worden gebruikt om spraak te verwerken en te analyseren om te helpen bij het herkennen van specifieke woorden in een spraakopname. Dit kan vervolgens worden gebruikt om tekst te genereren die de spraak weergeeft. Dit proces vereist echter training op basis van voorbeeldgegevens, zodat het neurale netwerk kan leren om de juiste tekst te genereren voor verschillende soorten spraak.

Een spraak-naar-tekst systeem kan dus worden gezien als een specifiek soort toepassing van een neuraal netwerk voor het doel van spraakherkenning en het genereren van tekst (MATLAB & Simulink, z.d.).

4.5 Hoe werkt een neuraal netwerk?

Neurale netwerken zijn geïnspireerd door biologische zenuwstelsels en werken ook met verschillende verwerkings lagen, met behulp van eenvoudige elementen die parallel werken. Het netwerk bestaat uit een invoer laag, één of meer verborgen lagen en een uitvoer laag. In elke laag zitten verschillende knooppunten van neuronen. De knooppunten in elke laag gebruiken de uitgangen van alle knooppunten in de vorige laag als invoer, zodat alle neuronen met elkaar verbonden zijn via de verschillende lagen. Elke neuron krijgt een gewicht toegewezen dat tijdens het leerproces wordt aangepast die de sterkte van het signaal van dat neuron vermindert of verhoogt (MATLAB & Simulink, z.d.).

De invoer laag bij een spraakherkenningssysteem is de spraak (audio). In de verborgen lagen wordt de audio omgezet naar tekst welke vervolgens wordt doorgestuurd naar de uitvoer laag.



4.6 Conclusie

Doordat er weinig data beschikbaar is vanuit rapportages uit de praktijk waarin medische termen, jargon en afkortingen staan is de keuze voor Transfer Learning het meest logisch. Op deze manier kan vanuit een goed presterend bestaand spraak-naar-tekst model verder gewerkt worden en met relatief weinig data aangetoond worden dat het mogelijk is om een bestaand model te trainen zodat deze woorden kan herkennen uit een eigen dataset.

Overige voordelen van Transfer Learning ten opzichte van het trainen van een nieuw model vanaf stratch zijn dat het zowel sneller als goedkoper is. Daarnaast kan een model na het opnieuw trainen door middel van Transfer Learning ook betrouwbaarder zijn dan wanneer een nieuw model vanaf stratch getraind wordt omdat er open source spraak-naar-tekst modellen beschikbaar zijn die al goed presteren. Dit is op te merken aan een lage Word Error Rate (WER).

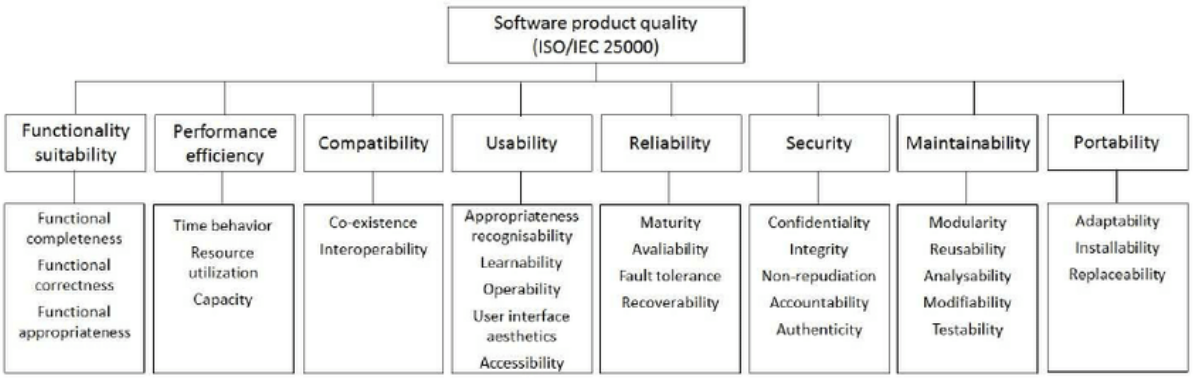
De Word Error Rate (WER) is de verhouding tussen fouten in een transcript en het totale aantal gesproken woorden. Een lagere WER in spraak-naar-tekst betekent een betere nauwkeurigheid bij het herkennen van spraak. Een WER van 20% betekent dat het transcript voor 80% nauwkeurig is.

5. Fase 3: Ontwerp en ontwikkeling

In de ontwerp en ontwikkelingsfase worden de functionele requirements eisen opgesteld waaraan het product moet voldoen. Voorafgaand de productrealisatie zal theoretisch onderzoek uitgevoerd worden die de deelvragen in de ‘ontwerp en ontwikkeling’ fase zullen beantwoorden. Het beantwoorden van deze deelvragen door middel van onderzoek is nodig om logische keuzes te kunnen maken voor de realisatie.

5.1 Requirements

In de ontwerp en ontwikkelingsfase worden de functionele en niet functionele eisen opgesteld waaraan het softwareproduct moet voldoen. De vereisten worden geprioriteerd volgens MoSCoW (Must haves, Should haves, Could haves, Won’t haves), gecategoriseerd volgens de ISO 25000-kwaliteitsstandaard van het software product kwaliteit model.



5.1.1 Niet functionele requirements

De niet functionele requirements zijn tijdens een meeting op 26 april 2022 (welke als doel had om de afstudeeropdracht te vormen) geformuleerd en geprioriteerd samen met de directie van Avinty Apps.

Nr.	Omschrijving	MoSCoW	ISO
NF1	De applicatie moet een losstaand project zijn	Must	Portability/ Installability
NF2	De applicatie moet integreerbaar kunnen zijn met de huidige software	Must	Portability/ Installability
NF3	De applicatie moet geïntegreerd kunnen worden in een Flutter app	Must	Portability/ Installability
NF4	De applicatie moet geïntegreerd kunnen worden in een web app	Must	Portability/ Installability
NF5	De applicatie moet volledig in het Nederlands te gebruiken zijn	Must	Usability/ Accessibility
NF6	Elk verzoek moet binnen 3 seconden worden verwerkt	Must	Performance efficiency/ Time behavior

NF7	Het transcriberen van audio moet gebeuren in real time	Must	Performance efficiency/ Time behavior
NF8	De applicatie moet commercieel inzetbaar kunnen zijn	Must	*Business requirement*
NF9	De applicatie moet werken op telefoons vanaf Android 8.0 en iOS 12.0	Should	Portability/ Installability
NF10	Het systeem moet een uptime hebben van 99.9%	Should	Reliability/ Availability
NF11	De applicatie kan meertalig zijn	Could	Usability/ Accessibility
NF12	De applicatie moet patenteerbaar kunnen zijn	Could	*Business requirement*

5.1.2 Functionele requirements

De functionele requirements zijn opgesteld op basis van de onderzoeksresultaten uit de concept- en haalbaarheidsfase. Deze requirements zijn geformuleerd en geprioriteerd samen bedrijfsbegeleider Mark Mooibroek.

Nr.	**Omschrijving**	**MoSCoW**	**ISO**
F1	Nederlandse spraak moet d.m.v. spraakinvoer omgezet worden naar tekst	Must	Functionality suitability/ Functional completeness
F2	Het spraakherkenning model moet medische termen kunnen herkennen gevoed door een eigen dataset	Must	Functionality suitability/ Functional completeness
F3	De server moet audio van meerdere clients tegelijk kunnen verwerken	Must	Functionality suitability/ Functional completeness
F4	Spraakinvoer moet gepauzeerd kunnen worden	Should	Functionality suitability/ Functional completeness
F5	Spraakinvoer moet hervat kunnen worden	Should	Functionality suitability/ Functional completeness
F6	De server moet spraakinvoer kunnen opslaan als audiobestand	Should	Functionality suitability/ Functional completeness
F7	Tekst moet vervangen kunnen worden door spraak d.m.v. het selecteren van tekst	Could	Functionality suitability/ Functional completeness
F8	Woorden moeten vervangen kunnen worden vanuit een lijst met suggesties gebaseerd op de spraakinvoer	Could	Functionality suitability/ Functional completeness

5.2 Dataverwerking

Er zijn verschillende voordelen en nadelen van het verwerken van data in de front-end of de back-end van een spraakherkenning applicatie.

5.2.1 Front-end

Bij onderstaande vergelijking is uitgegaan van een Flutter applicatie waarbij de audio transcriptie module volledig is ingebouwd in de applicatie en te gebruiken is zonder internetconnectie. De keuze voor Flutter is gemaakt doordat in de niet-functionele vereisten bepaald is dat de applicatie geïntegreerd moet kunnen worden met een Flutter- en webapplicatie. Bij het realiseren van de applicatie in Flutter wordt voldaan aan beide vereisten omdat een Flutter applicatie ook in web kan draaien (Flutter, z.d.).

Voordelen	**Nadelen**
De dataverwerking wordt uitgevoerd op de client-side, wat betekent dat de server minder belast wordt en de response-tijd sneller kan zijn.	De client heeft mogelijk geen toegang tot alle nodige en nieuwe data, wat kan betekenen dat de front-end beperkt is in wat het kan bereiken.
Dataverwerking in de front-end is veiliger doordat audio waarin gevoelige patiëntgegevens in staat niet over een netwerk verzonden hoeft te worden.	De client moet in staat zijn om de benodigde berekeningen uit te voeren, wat kan betekenen dat de client-side hardware en software beperkter zijn dan die op de server. Dit kan leiden tot minder efficiënte dataverwerking. Dit kan ook resulteren in dat de applicatie zelf erg groot wordt.

5.2.2 Back-end

Bij onderstaande vergelijking is uitgegaan van een API server die de audio transcriptie verwerkt.

Voordelen	**Nadelen**
De back-end heeft doorgaans toegang tot meer data en resources, wat kan leiden tot efficiëntere dataverwerking.	De server kan een hogere belasting ervaren als er veel dataverwerking plaatsvindt, wat kan leiden tot tragere prestaties.
De backend kan worden ontworpen om grote hoeveelheden data efficiënt te verwerken en te schalen (door bijvoorbeeld VRAM geheugen toe te voegen) naarmate de vraag toeneemt.	Dataverwerking in de back-end kan leiden tot langere responstijden omdat de dataverwerking op de server plaatsvindt en de resultaten naar de client verzonden moeten worden.

In het algemeen is het afhankelijk van de specifieke behoeften van de toepassing en het gewenste niveau van prestaties welke optie het beste is. Soms kan het zinvol zijn om een combinatie van front-end en back-end dataverwerking te gebruiken om het beste van beide werelden te combineren.

Na het analyseren van de voor- en nadelen van beide opties is de keuze gemaakt om de audio transcriptie in de back-end te laten plaatsvinden. Bij deze keuze weegt vooral zwaar dat de applicatie erg groot wordt als de gehele transcriptie module ingebouwd wordt en te gebruiken is zonder internetconnectie. Daarnaast weegt in deze keuze de schaalbaarheid en de beschikbare resources bij het verwerken van data in de back-end ook zwaar mee.

5.3 Applicatie infrastructuur

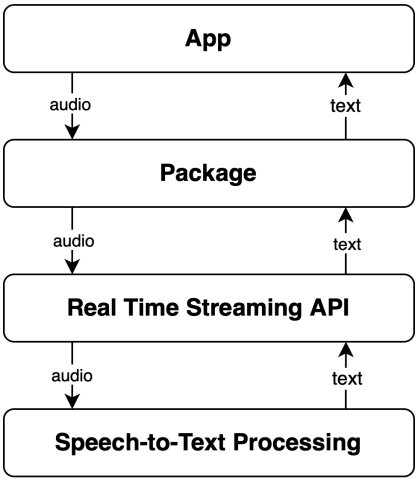
De applicatie bestaat uit 4 verschillende onderdelen waaronder:

- **App (Client):** in dit prototype een Flutter-applicatie die audio kan streamen vanuit de ingebouwde microfoon naar de server.

- **Package (Plugin):** in dit prototype een Flutter Package die de connectie en communicatie met de server regelt. (de code die te maken heeft met server connectie en communicatie zit in een losse Flutter package zodat deze code makkelijk overdraagbaar en implementeerbaar is bij de huidige software).

- **API (Server):** ontvangt de audio bytes van de plugin via een real time streaming protocol en stuurt deze door naar de audio processing laag.

- **Audio Processing:** op deze laag vindt de audio transcriptie plaats. Het transcriberen gebeurt door een Machine Learning model welke getraind is op basis van medische dossiers. Het model verwacht audio als input en geeft tekst terug welke teruggestuurd wordt via de server en plugin naar de client (app).



5.4 Software architectuur

Avinty ontwikkeld en onderhoud een eigen Flutter boilerplate die gebaseerd is op een eigen variant van clean architecture.

Clean architecture is een software-ontwerpfilosofie die de elementen van een ontwerp scheidt in verschillende niveaus. Een belangrijk doel van deze architectuur is om ontwikkelaars een manier te bieden om code zo te organiseren dat de business logic gescheiden gehouden wordt van de gebruikersinterface en de data waaronder API calls en bijvoorbeeld database streams vallen (Clean architecture, 2019).

De hoofdregel van clean architectuur is dat code-afhankelijkheden alleen van de buitenste niveaus naar binnen kunnen bewegen. Code op de binnenste lagen kan geen kennis hebben van functies op de buitenste lagen. De variabelen, functies en klassen die in de buitenste lagen bestaan, kunnen niet worden genoemd in de daarbinnen gelegen niveaus (Clean architecture, 2019).

De richtlijnen van deze clean architecture vormen de basis van hoe het Flutter project gebouwd en ingericht wordt. De richtlijnen van clean architecture bestaan uit 4 onderdelen waaronder: de data-, domein- en UI laag, en de core.

- **Core:** alle lagen zijn afhankelijk van core. Core wordt gebruikt om alle lagen te registeren en de juiste implementaties voor bepaalde klassen te krijgen. Daarnaast wordt met Widget Tests de UI laag getest en de worden de services en databronnen 'gemocked'.

- **Data:** de data laag bevat services, databronnen en repositories. Componenten in de datalaag communiceren met de API en worden via repositories opengesteld aan de domeinlaag.

- **Domein:** de domeinlaag bevat models, converters en viewmodels. De domeinlaag wordt gebruikt om de business logic te verwerken in de data verkregen vanuit de data laag.

- **UI:** de UI-laag bevat assets zoals images, vertalingen en fonts. Maar ook alle widgets, hooks en routing componenten. De UI haalt data die weergegeven moet worden op vanuit de domeinlaag, en communiceert daardoor niet vanuit zichzelf naar buiten.

Een visuele representatie van deze architectuur kan gevonden worden in Appendix C.

5.5 Real Time Streaming

Om audio te streamen van de client (Flutter app) naar de server (API) moet gebruik gebruik gemaakt worden van een netwerkprotocol die wordt gebruikt om gegevens tussen computers en andere apparaten uit te wisselen. Hiervoor kan gebruik gemaakt worden van Data Streaming protocollen waaronder bijvoorbeeld MQTT, gRPC of Websockets.

- **MQTT:** is een licht gewicht publish/subscribe-protocol ontworpen voor verbindingen met lage bandbreedte en hoge latentie. Één van de voordelen van MQTT bij audiostreaming is dat het efficiënt is in bandbreedte en stroomverbruik, omdat het alleen gegevens verzendt wanneer dat nodig is in een compact binair formaat. MQTT is ontworpen om betrouwbaar te zijn en zorgt ervoor dat gegevens zelfs over onstabiele netwerken worden afgeleverd.

- **Websockets:** is gericht op real-time bidirectionele verbindingen waardoor het een goede keuze kan zijn voor audio streaming voor een spraakherkenning applicatie. Websockets zou een constante verbinding tussen client en server kunnen opzetten, wat nodig is voor een spraakherkenning applicatie die in real-time werkt.

- **gRPC:** is gericht op hoge prestaties en biedt ondersteuning voor verschillende talen. Het maakt gebruik van het HTTP/2-protocol voor transport van berichten, wat kan helpen bij het verminderen van latentie. Daarom zou gRPC een goede keuze kunnen zijn voor audio streaming voor een spraakherkenning applicatie.

Type	MQTT	Websockets	GRPC
1000 messages (1.5KB)	142 ms	16 ms	10 ms
10000 messages (1.5KB)	1326 ms	127 ms	40 ms
100000 messages (1.5KB)	2613 ms	911 ms	219 ms
1000 messages (4KB)	250 ms	26 ms	10 ms
10000 messages (4KB)	1696 ms	208 ms	38 ms
100000 messages (4KB)	6880 ms	1767 ms	257 ms

Bovenstaande tabel laat zien hoeveel tijd het kost om berichten met bepaalde groottes en aantallen te streamen via MQTT, Websockets en gPRC. Deze resultaten laten duidelijk zien dat gRPC wint vanwege de aanhoudende verbinding en het protobuf-gegevensformaat, dat licht van gewicht is (Kannappa, 2021).

5.6 Spraakherkenningssystemen

In de haalbaarheidsfase is geconcludeerd dat d.m.v. Transfer Learning bestaande modellen getraind kunnen worden door middel van minder gelabelde gegevens door populaire modellen te hergebruiken die al zijn getraind op grote gegevenssets. Hierdoor is minder trainingstijd en rekenkracht vereist in vergelijking met het trainen van een nieuw model vanaf scratch (MATLAB & Simulink, z.d.).

Daarom worden populaire open-source spraakherkenningssystemen waaronder: Kaldi, DeepSpeech, Whisper en Wav2Letter met elkaar vergeleken en wordt er een keuze gemaakt voor een systeem gebaseerd op eisen passend bij het projectdoel (Foster, 2022).

- **Kaldi:** is een spraakherkenningssysteem dat is ontworpen om te worden aangepast aan verschillende talen. Kaldi is een populaire keuze voor spraakherkenning onderzoek en ontwikkeling en wordt gebruikt in veel commerciële producten. Kaldi biedt een uitgebreide set van algoritmen en is makkelijk te integreren met andere software (Foster, 2022). Het Nederlandse Kaldi model heeft een Word Error Rate van 5.2% (Tejedor-García, z.d.).

- **Whisper:** is een spraakherkenningssysteem gericht op het bieden van een gemakkelijk te gebruiken interface voor spraakherkenning onderzoek en ontwikkeling. Whisper biedt ook een aantal geavanceerde spraakherkenning algoritmen en wordt vaak gebruikt in combinatie met andere software (Foster, 2022). Het grootste meertalige model van Whisper heeft een Word Error Rate van 7.1% (Radford, 2022).

- **DeepSpeech:** is een spraakherkenning model ontwikkeld door Mozilla. Het maakt gebruik van een end-to-end deep learning model voor spraakherkenning en is specifiek gericht op het verbeteren van de prestaties op langere audiobestanden. DeepSpeech is gemakkelijk te integreren met andere software en wordt vaak gebruikt in commerciële producten (Foster, 2022). DeepSpeech heeft een Word Error Rate van 8.2% (Röpke, 2019).

- **Wav2Letter:** is een spraakherkenning model ontwikkeld door Facebook. Het is gericht op het verbeteren van de prestaties van spraakherkenning op langere audiobestanden en maakt gebruik van een end-to-end diep leer model. Wav2Letter is een populaire keuze voor spraakherkenning onderzoek en wordt vaak gebruikt in combinatie met andere open-source software (Foster, 2022). Wav2Letter heeft een Word Error Rate van 10.6% (Ycombinator, z.d.).

De keuze voor open-source verleent Avinty het recht om het systeem voor elk doel te gebruiken, te bestuderen, te wijzigen en te distribueren.

In het algemeen zijn Kaldi, Whisper, DeepSpeech en Wav2Letter allemaal krachtige spraakherkenningssystemen met hun eigen kenmerken en voordelen. Welk systeem het beste is hangt af van het projectdoel, de specifieke behoeften en eisen. Daarom zijn de volgende vereisten opgesteld om een goede keuze te kunnen maken (Foster, 2022):

5.6.1 Criteria

De eerste stap bij het kiezen van een spraakherkenningssysteem is het bedenken van criteria om elk systeem te beoordelen.

Omschrijving	**Verantwoording**
Het spraakherkenningssysteem moet goed gedocumenteerd zijn	Er wordt gebruik gemaakt van complexe technologieën, daardoor is een goede documentatie essentieel in het begrijpen van het systeem.
Er moeten fine-tuning (het aanpassen van modellen mogelijkheden zijn	Dit maakt het mogelijk om een model te trainen op een zelf samengestelde dataset
Real time audio verwerking moet mogelijk zijn	Dit maakt het mogelijk om ingesproken tekst in real time te zien verschijnen in het invoerveld
De Word Error Rate moet tussen de 5% en 10% liggen	Een WER van 5-10% wordt gezien als goede kwaliteit en is klaar voor gebruik (Microsoft, 2022)
Het model moet Nederlandse taal ondersteunen	De rapportages worden door zorgprofessionals in het Nederlands geschreven
Het model moet commercieel inzetbaar kunnen zijn	De mogelijkheid om het model commercieel te mogen gebruiken biedt Avinty de mogelijkheid om geld te verdienen met de software
Het model moet patenteerbaar kunnen zijn	Patenten bieden Avinty de mogelijkheid om extra winst te kunnen behalen

5.6.2 Confrontatie- en Pugh matrix

Het doel is om een objectieve keuze te maken tussen vier populaire open-source spraakherkenningssystemen. De Pugh-matrix een uitstekende methode om een objectieve keuze te maken. Door te focussen op criteria op een passend detailniveau, kunnen de resulterende beslissingen worden bepaald door feiten in plaats van door voorkeuren en meningen (Frey, 2008, blz. 57).

De Pugh-matrix bestaat uit verschillende criteria en gewichtsfactoren. Een goede manier om deze gewichtsfactoren te bepalen is door een confrontatiematrix te gebruiken (Jenčo & Černák, 2019, blz. 76). In de confrontatiematrix worden de vereisten in zowel de rijen als de kolommen van de tabel geplaatst en vervolgens tegen elkaar opgezet. Per knooppunt wordt aangegeven of de criteria in de rij belangrijker is dan de criteria in de kolom. Wanneer dat het geval is wordt er een 1 op het knooppunt geplaatst, wanneer dit niet zo is wordt er een 0 neergezet. Vervolgens worden deze scores bij elkaar opgeteld en komen hier de wegingsfactoren uit.

5.6.3 Confrontatiematrix

Volgens de confrontatiematrix is de open-source codebase het belangrijkste criterium, gevolgd door fine-tuning mogelijkheden. Deze criteria zijn het belangrijkst, want het projectdoel is sterk afhankelijk van wat er beschikbaar is gemaakt door de uitgevers en hoeveel informatie er te vinden is over het fine-tunen van de specifieke modellen. De ingevulde confrontatiematrix bepaald de gewichtsfactoren voor de Pugh-matrix.

	Documentatie	Fine-tuning	Word Error Rate	Nederlands	Real time audio	Commercieel inzetbaar	Patenteerbaar	**Gewichtsfactor**
Documentatie	X	1	1	1	1	1	1	7
Fine-tuning	0	X	1	1	1	1	1	6
Word Error Rate	0	0	X	1	1	1	1	5
Nederlands	0	0	0	X	1	1	1	4
Real time audio	0	0	0	0	X	1	1	3
Commercieel inzetbaar	0	0	0	0	0	X	1	2
Patenteerbaar	0	0	0	0	0	0	X	1

5.6.4 Pugh matrix

Zodra de confrontatiematrix ingevuld is kan de Pugh-matrix opgesteld worden. Elk spraakherkenningssysteem wordt afzonderlijk beoordeeld op basis van de criteria. Vervolgens worden deze cijfers vermenigvuldigd met de gewichtsfactor om de totale scores te kunnen bepalen. Deze cijfers zijn relatief ten opzichte van de overige spraakherkenningssystemen en zijn gebaseerd op informatie verkregen vanuit desk research. De scores in de Pugh matrix worden toegelicht in de volgende paragraaf.

Criteria	Gewichtsfactor	Kaldi		Whisper		DeepSpeech		Wav2Letter	
		Cijfer	Totaal	Cijfer	Totaal	Cijfer	Totaal	Cijfer	Totaal
Documentatie	7	9	63	8,5	59,5	8,5	59,5	8	56
Fine-tuning	6	8	48	10	60	8	48	7,5	45
Word Error Rate	5	9	45	8,5	42,5	8	40	7,5	37,5
Nederlands	4	8,5	34	8,5	34	7	28	7	28
Real time audio	3	9	27	8,5	25,5	7.5	22,5	6	18
Commercieel inzetbaar	2	10	20	10	20	10	20	10	20
Patenteerbaar	1	10	10	8	8	10	10	1	1
		Totaal: 247		**Totaal: 249,5**		**Totaal: 228**		**Totaal: 205,5**	

5.6.5 Score uitleg

Om de redenering achter de scores in de Pugh-matrix te begrijpen, wordt elke criteria doorgenomen en zal de redenering toegelicht worden.

1. **Documentatie**

De scores bij ‘documentatie’ zijn naast de beschikbare publicaties afhankelijk van de kwaliteit van documentatie waarin aspecten zoals beschikbare code voorbeelden, aanwezigheid van gedetailleerde uitleg, maar ook structuur meeweegt in de beoordeling. Hoewel de documentatie bij alle systemen relatief goed is, is deze bij Kaldi diepgaander vergeleken met de rest. Whisper is een redelijk nieuw systeem gepubliceerd in september 2022. Overigens heeft het bedrijf achter Whisper genaamd OpenAI veel bekendheid gekregen door eerdere publicaties waardoor er ook veel aandacht wordt besteed aan publicaties voor Whisper waaronder real time audio processing projecten en stap-voor-stap handleidingen om Whisper modellen te trainen. In het geval van DeepSpeech en Wav2Letter is de documentatie kwalitatief minder.

2. **Fine-tuning**

De scores bij ‘fine-tuning’ zijn afhankelijk van de publicaties over het trainen van de modellen. Het bedrijf Hugging Face heeft een zeer uitgebreide publicatie genaamd “Fine-Tune Whisper For Multilingual ASR with Transformers” waarin stap-voor-stap uitgelegd wordt hoe het fine-tunen werkt voor Whisper, waarin ook de theorie erachter het trainen behandeld wordt. Voor Kaldi zijn er meerdere tutorials beschikbaar, maar deze lijken kwalitatief minder te zijn. In het geval van DeepSpeech en Wav2Letter is er aanzienlijk minder informatie te vinden over het trainen van deze modellen.

3. **Word Error Rate**

De scores bij ‘word error rate’ zijn afhankelijk van de Word Error Rate (WER) van de verschillende systemen. Kaldi heeft een WER van 5.2%, Whisper heeft een WER van 7.1%, DeepSpeech heeft een WER van 8.2% en Wav2Letter heeft een WER van 10.6%.

4. **Nederlands**

De scores bij ‘nederlands’ zijn afhankelijk van of de spraakherkenningssystemen modellen hebben die de Nederlandse taal ondersteunen. Zowel Kaldi als Whisper hebben modellen die Nederlands ondersteunen. In het geval van DeepSpeech en Wav2Letter is er vrij weinig te vinden over de beschikbaarheid van Nederlandse spraakherkennings modellen.

5. **Real time audio**

De scores bij ‘real time audio’ zijn afhankelijk van of de modellen zelf real time audio ondersteunen. Kaldi heeft standaard ondersteuning voor real time audio. Whisper heeft dit niet standaard in het model zitten, maar er zijn publicaties die real time audio processing mogelijk maakt. Bij DeepSpeech en Wav2Letter lijkt hetzelfde het geval te zijn, overigens lijkt het implementeren van real time audio processing lastiger te zijn bij deze modellen door het aantal beschikbare informatie die hierover gepubliceerd is.

6. **Commercieel inzetbaar**

De scores bij ‘commercieel inzetbaar’ zijn afhankelijk van de licenties die gekoppeld zijn aan de verschillende spraakherkenningssystemen. Kaldi NL maakt gebruik van de ‘Apache License 2.0’, DeepSpeech gebruikt de ‘Mozilla Public License Version 2.0’, Whisper maakt gebruik van de ‘MIT License en Wav2Letter gebruikt de ‘BSD License’ als licentie. Bij al deze licenties is het commercieel gebruiken van de software toegestaan.

7. **Patenteerbaar**

De scores bij ‘patenteerbaar’ zijn afhankelijk van de licenties. Kaldi NL maakt gebruik van de ‘Apache License 2.0’ licentie wat betekent dat Kaldi NL aangepast, gedistribueerd en commercieel gebruikt mag worden, ook staat de licentie patent registratie toe. DeepSpeech maakt gebruik van de ‘Mozilla Public License Version 2.0’ waarbij de rechten op hetzelfde neerkomen als bij de ‘Apache License’. Whisper maakt gebruik van de ‘MIT License’ wat betekent dat de software aangepast, gedistribueerd en commercieel gebruikt mag worden. Overigens is er niks gedocumenteerd over of patent registratie toegestaan is, wat in dit geval positief kan zijn. Wav2Letter maakt gebruik van een BSD License waarbij patent registratie niet toegestaan is.

5.6.6 Conclusie

Whisper krijgt de meeste punten in de Pugh-matrix (Frey, 2008, blz. 57). Dit resultaat bepaalt de keuze van het spraakherkenningssystemen en het mogelijke model waarop verder getraind zal worden.

5.7 Whisper

De modellen van Whisper zijn beschikbaar in verschillende types met elk verschillende groottes. Er zijn transcriptie modellen die alleen Engels ondersteunen, en modellen die meerdere talen ondersteunen. Bij de meertalige modellen wordt de taal automatisch herkend, ook is het mogelijk om de taal hardcoded aan te geven als er bijvoorbeeld gebruik gemaakt wordt van één taal.

De meertalige modellen, benodigd voor deze opdracht bestaan uit vijf verschillende groottes waaronder: tiny, base (standaard), small, medium en large - waarbij het model per grootte steeds nauwkeuriger wordt (Radford, 2022).

Bij het kiezen van de model grootte wordt gekeken naar de vereiste rekenkracht, de snelheid relatief tot de andere modellen, de Word Error Rate van de modellen op de Common Voice 9 dataset en de Whisper Inference Time (O'Connor, 2022).

Model	Parameters	Required VRAM	Relative speed	WER (%) on Common Voice 9
tiny	39 M	~1 GB	~32x	43.6
base	74M	~1 GB	~16x	29.5
small	244 M	~2 GB	~6x	14.2
medium	769 M	~5 GB	~2x	8.0
large	1550 M	~10 GB	~1x	7.1

Een WER van 5-10% wordt gezien als goede kwaliteit en is klaar voor gebruik. Een WER van 20% is acceptabel, maar aanvullende training wordt geadviseerd. Een WER van 30% of meer betekent slechte kwaliteit en vraagt om maatwerk en training (Microsoft, 2022).

Om tijd te besparen tijdens het testen en het finetunen van een model is er gekozen om het meertalige ‘small’ transcriptie model van Whisper te gebruiken. De vereiste VRAM t.o.v. de snelheid in vergelijking tot de overige modellen is bij het smalle model het best. Het enige nadeel van het smalle model is de Word Error Rate van 14.2% t.o.v. van het ‘medium’ model 8.0%. Overigens is deze WER omlaag te krijgen tijdens het fine-tunen.

Mocht de WER tijdens het fine-tunen niet onder de 10% uitkomen, maar het projectdoel om een transcriptie model medische termen te laten herkennen vanuit een eigen dataset wel behaald wordt, kan er later altijd nog gekozen worden om het medium, of grootte model te gebruiken in een productieomgeving.

5.8 Fine-tuning Whisper

Het doel van het finetunen van Whisper is het kunnen aantonen dat een eigen getraind model medische termen, jargon en afkortingen kan herkennen die de modellen van Whisper niet uit zichzelf herkend. Om dit doel te kunnen bereiken moet het model opnieuw getraind worden d.m.v. een eigen dataset. Dit heet ook wel het finetunen van een spraakherkenning model.

Bij het finetunen van het smalle Whisper model wordt gekeken hoe een model zo efficiënt mogelijk getraind kan worden. Dit betekend het selecteren van de juiste hardware en infrastructuur waarbij aspecten zoals tijdsduur en kosten meegenomen worden in de beslissing.

5.8.1 Benodigdheden

Het model wordt getraind met behulp van de blogpost genaamd “Fine-Tune Whisper With Transformers and Streaming Mode” door het bedrijf Hugging Face. Deze blog geeft een diepgaande uitleg van het Whisper-model, en de theorie achter fine-tuning, met bijbehorende code om de stappen voor datapreparatie en fine-tuning uit te voeren.

De benodigdheden volgens deze blogpost is een sterke GPU voor het trainen van een model, hierbij wordt de GPU aangeraden die komt bij een Google Colab Pro of Pro+ abonnement. Dit is een NVIDIA A100 SXM 4 kaart met 40 GB aan videogeheugen.

Google Colab Pro kost maandelijks €9,25, de Pro+ versie van het abonnement kost maandelijks €42,25. De Pro+ variant komt met 5x meer compute units en de mogelijkheid om scripts op de achtergrond te laten draaien, zelfs wanneer de browser gesloten wordt. Het trainen van een model kost ongeveer 14 compute units per uur en duurt gemiddeld 18 uur voor het trainen van een model op de Nederlandse Common Voice 11 dataset.

Daarnaast is een gelabelde dataset vereist zodat het model getraind kan worden door middel van supervised learning. Supervised learning is een type machine learning waarbij een algoritme getraind wordt met behulp van een dataset waarvan de output al bekend is. Het algoritme leert hierdoor een relatie tussen de input en de output, zodat het in staat is nieuwe data te classificeren of te voorspellen (MATLAB & Simulink, z.d.).

5.8.2 Datasets

Gegevens voor een dataset kunnen op meerdere manieren gesplitst worden, elk type split resulteert in verschillende prestaties. De keuze voor een bepaalde split is afhankelijk van het projectdoel en de hoeveelheid beschikbare data.

De drie meest gebruikelijke manieren van het splitsen van data zijn als volgt:

1. Alleen een training dataset;
2. Een training- en validatie dataset;
3. Een training-, validatie- en test dataset.

5.8.2.1 Training data set

Wanneer de volledige gegevensset wordt gebruikt om het model te trainen is er geen nieuwe data beschikbaar om mee te valideren of te testen. Het model wordt getraind met behulp van de volledige dataset en test de prestaties van het model op de willekeurige gegevens uit de volledige trainings dataset. Het resultaat hiervan is dat het model een sterke voorkeur voor de dataset krijgt. Een dergelijk model zal hoogstwaarschijnlijk niet kunnen generaliseren op ongeziene gegevens, tenzij de dataset die voor training wordt gebruikt, de hele populatie vertegenwoordigt (Kumar, 2021).

5.8.2.2 Training- en validatie dataset

Wanneer de gegevensset in tweeën gesplitst wordt, wordt de ene splitsing een trainings gegevensset en de andere splitsing een validatie gegevensset genoemd. Het model wordt getraind met behulp van de trainings gegevensset en evalueert de model prestaties met behulp van de validatie gegevensset.

Over het algemeen wordt de dataset voor training en validatie opgesplitst in een verhouding van 80:20. Zo wordt 20% van de gegevens gereserveerd voor validatie. De verhouding verandert op basis van de grootte van het aantal gegevens. In het geval dat het aantal gegevens erg groot is, wordt ook wel gekozen een 90:10 gegevens splitsing ratio waarbij de validatie gegevensset 10% van de gegevens vertegenwoordigt.

Wanneer de dataset wordt getraind met behulp van de trainings dataset en wordt geëvalueerd op de validatie dataset, presteert het model veel beter dan het eerdere model dat is getraind met de volledige dataset (Kumar, 2021).

5.8.2.3 Training-, validatie- en test dataset

Wanneer een dataset in drie delen gesplitst wordt resulteert dit in een trainings-, validatie- en test dataset. Het model wordt getraind met behulp van de trainings gegevensset en de model prestaties worden beoordeeld met behulp van de validatie gegevensset. De model prestaties worden geoptimaliseerd met behulp van trainings- en validatie dataset. Ten slotte wordt de model generalisatie getest met behulp van de test gegevensset.

De test gegevensset blijft verborgen tijdens de modeltraining en de evaluatiefase van de model prestaties. Over het algemeen wordt de dataset voor training, validatie en testen opgesplitst in een verhouding van 70:20:10. 10% van de dataset kan gereserveerd worden als testdata voor het testen van de model prestaties.

Modellen hebben een grotere kans om te generaliseren op een ongeziene dataset dan de twee eerder genoemde gevallen (Kumar, 2021).

5.8.2.4 Minimale hoeveelheid audio

De minimale hoeveelheid audio die nodig is om een machine learning-model voor spraak-naar-tekst te trainen, kan variëren, afhankelijk van de complexiteit van het model en de diversiteit van de spraakgegevens die voor training wordt gebruikt. Meestal is een dataset van ten minste een paar honderd uur aan audio nodig om een eenvoudig spraak-naar-tekst model te trainen.

Het trainen van machine learning-modellen met een kleinere dataset kan acceptabele resultaten opleveren, maar met meer data kun je de prestaties van het model verbeteren, vooral wat betreft het herkennen van spraak van verschillende accenten en dialecten (Quora, z.d.).

5.8.3 Model training op Common Voice 11

Om te valideren dat het trainen van een model mogelijk is en ook succesvol werkt via Google Colab is de eerste stap het trainen van een model op een veel gebruikte publieke dataset genaamd Common Voice 11.

De resultaten van het trainen van het smalle Whisper model op de Nederlandse data van de Common Voice 11 dataset kan gevonden worden in het hoofdstuk ‘Fase 4: Testen’.

5.8.4 Model training op GGZ behandelplan

Zodra het trainen van het smalle Whisper model op de Nederlandse data van de Common Voice 11 dataset succesvol is, kan er gekeken worden hoe een model getraind kan worden door middel van een eigen dataset.

De resultaten van het trainen van het smalle Whisper model op een eigen dataset bestaande uit teksten vanuit een GGZ behandelplan kan gevonden worden in het hoofdstuk ‘Fase 4: Testen’.

5.8.4.1 Samenstelling van de dataset

Bij het samenstellen van de dataset is bij GGZ een behandelplan uit de praktijk opgevraagd welke vervolgens geanonimiseerd is. Op basis van dit behandelplan zijn zinnen opgesteld waar de ingebouwde spraakherkenning van Apple, Samsung en Google moeite mee hebben. Deze zinnen bevatten namen van bedrijven, afkortingen, medische termen en jargon.

5.8.4.2 Dataset voor Whisper Fine-Tuning

De teksten uit deze dataset zijn ingesproken door verschillende mannelijke en vrouwelijke collega’s waaronder mijzelf, in totaal bestaat deze dataset uit 14 minuten aan audio waaronder 20 zinnen welke ingesproken zijn door 7 verschillende sprekers. Doordat de dataset bestaat uit 14 minuten aan audio en 7 sprekers is vast te stellen dat deze dataset erg beperkt is. Ter vergelijking, de Nederlandse data van Common Voice heeft 113 uur aan audio waaronder meer dan 250.000 zinnen ingesproken door meer dan 1500 verschillende sprekers (Mozilla Common Voice, z.d.). Link naar de GGZ dataset

Overigens is het doel van het prototype het kunnen aantonen dat een Machine Learning model woorden kan herkennen welke het model eerst niet herkend na het trainen van het model op een eigen dataset. Het valideren van de mogelijk hiervan door middel van een prototype met het bijbehorende onderzoek is het belangrijkste doel van Avinty. De beperkte dataset staat daarom het behalen dan het projectdoel niet in de weg.

Omdat de dataset relatief klein is, maar het model wel gevalideerd moet worden wordt er gebruik gemaakt van een 80:20 split. Dit betekend dat 80% van de data gebruikt wordt om het model mee te trainen, de overige 20% van de gegevens gereserveerd voor validatie.

Het GGZ behandelplan kan gevonden worden in Appendix D.

De dataset voor Whisper Fine-Tuning kan gevonden worden in Appendix E.

5.9 Prototype

Het prototype bestaat uit een Flutter App (client) waarmee een gebruiker audio kan streamen vanuit de ingebouwde microfoon naar de server door middel van een zelf ontwikkelde Flutter package. Deze Flutter package regelt de connectie en communicatie met de server via het gRPC netwerkprotocol. De API (server) is een Python server die real time audio ontvangt als bytes en deze doorstuurt naar het zelf getrainde Whisper model. Het Whisper model zet deze audio om in tekst en stuurt deze tekst via de server en de Flutter package terug naar de app (client) welke vervolgens wordt weergegeven aan de gebruiker.

6. Fase 4: Testen

In de testfase wordt het prototype uitgebreid getest om er zeker van te zijn dat er geen fouten zijn gemaakt tijdens de ontwikkeling en dat het product voldoet aan de eisen die opgesteld zijn in de eerdere fases. Daarnaast zal de nauwkeurigheid van het model worden getest en gekeken worden of er punten zijn die verbeterd moeten worden.

6.1 Word Error Rate

De nauwkeurigheid van de spraak-naar-tekst technologie wordt uitgedrukt in de Word Error Rate (WER). Deze maat geeft aan welk percentage van de uitgesproken woorden foutief is omgezet naar tekst. Een model die niet gespecialiseerd is op het taalgebruik in de GGZ heeft al snel een WER van 30% wanneer een standaard model getest wordt op teksten in een behandelplan. Dit betekent dat als een begeleider een tekst inspreekt gemiddeld 3 op de 10 woorden niet correct zijn. Dit zorgt ervoor dat zorgprofessionals kostbare tijd kwijt zijn aan het corrigeren van de rapportage (Attendi, z.d.).

Een WER van 5-10% wordt gezien als goede kwaliteit en is klaar voor gebruik. Een WER van 20% is acceptabel, maar aanvullende training wordt geadviseerd. Een WER van 30% of meer betekent slechte kwaliteit en vraagt om maatwerk en training (Microsoft, 2022).

6.2 Whisper

Het spraakherkenning model wordt apart van het prototype getest door middel van het berekenen van de Word Error Rate op de validatieset na het trainen van het model.

6.2.1 Common Voice 11

Het trainen van het smalle Whisper model op de Nederlandse versie van de Common Voice 11 dataset via Google Colab was succesvol en resulteert in een WER van 11.6177.
Link naar het Common Voice 11 model (parla-nl-v1)

6.2.2 GGZ behandelplan

Het trainen van het smalle Whisper model op een eigen dataset gebaseerd op een GGZ behandelplan via Google Colab was succesvol en resulteert in een WER van 9.82.
Link naar het GGZ model (parla-nl-v2)

6.3 Test cases

Het prototype wordt apart van het spraakherkenning model getest door middel van verschillende testcases. Het doel van deze testen is het controleren dat voldaan wordt aan de opgestelde niet functionele- en functionele requirements in de ontwerp- en ontwikkelingsfase.

De test cases voor het prototype kunnen gevonden worden in Appendix F.

6.4 Traceability matrix

Om er zeker van te zijn dat elke requirement wordt getest is er een traceability matrix opgesteld welke gebaseerd is op de test cases. In de onderstaande tabel wordt door middel van deze traceability matrix elke testcase gekoppeld aan een functionele requirement wat ervoor zorgt dat elke requirement getest wordt.

	TC 1	TC 2	TC 3	TC 4	TC 5	TC 6	TC 7	TC 8
F1	✅							
F2		✅						
F3			✅					
F4				✅				
F5					✅			
F6						✅		
F7							❌	
F8								❌

6.5 Niet functionele requirements

Doordat niet alle functionele requirements getest kunnen worden door middel van testcases is de volgende tabel opgesteld welke laat zien waarom een requirement wel/niet behaald is door middel van een verantwoording.

Nr.	**Omschrijving**	**MoSCoW**	**Status**	**Verantwoording**
NF1	De applicatie moet een losstaand project zijn	Must	✅	Het prototype is een losstaand project
NF2	De applicatie moet integreerbaar kunnen zijn met de huidige software	Must	✅	Doordat de lagen in de infrastructuur goed opgesplitst zijn is het integreren makkelijk en mogelijk
NF3	De applicatie moet geïntegreerd kunnen worden in een Flutter app	Must	✅	De front-end (client) is een Flutter applicatie
NF4	De applicatie moet geïntegreerd kunnen worden in een web app	Must	✅	De front-end (client) is een Flutter applicatie welke op web ook werkt
NF5	De applicatie moet volledig in het Nederlands te gebruiken zijn	Must	✅	De applicatie is getest en volledig te gebruiken in het Nederlands
NF6	Elk verzoek moet binnen 3 seconden worden verwerkt	Must	✅	De audiotranscriptie gebeurt in real time, voor de rest zijn er geen verzoeken

NF7	Het transcriberen van audio moet gebeuren in real time	Must	✅	De audiotranscriptie gebeurt in real time
NF8	De applicatie moet commercieel inzetbaar kunnen zijn	Must	✅	Door de MIT License (Whisper) is de applicatie commercieel inzetbaar
NF9	De applicatie moet werken op telefoons vanaf Android 8.0 en iOS 12.0	Should	✅	De applicatie is getest op verschillende toestellen op verschillende OS versies
NF10	Het systeem moet een uptime hebben van 99.9%	Should	✅	Met een betrouwbare host kan het systeem een uptime van 99.9% behalen
NF11	De applicatie kan meertalig zijn	Could	✅	Het spraakherkenning model waarop getraind is, is meertalig
NF12	De applicatie moet patenteerbaar kunnen zijn	Could	✅	Door de MIT License (Whisper) is de applicatie patenteerbaar

6.6 Nice to have functionaliteiten

De traceability matrix en de tabel hierboven concluderen dat aan alle ‘must have’ en ‘should have’ functionaliteiten zijn voldaan. Helaas was er geen ruimte voor bepaalde ‘could have’ functionaliteiten in de planning. Het prototype is overigens volledig werkend doordat alle ‘must have’ en ‘should have’ functionaliteiten zijn gerealiseerd. Alleen bepaalde ‘extra’ functionaliteiten die de app afmaken zitten er helaas niet in. Hierover is meer te lezen in de verbeterfase. De functionaliteiten die hieronder vallen zijn als volgt:

Nr.	**Omschrijving**	**MoSCoW**	**Status**
F8	Tekst moet vervangen kunnen worden door spraak d.m.v. het selecteren van tekst	Could	❌
F9	Woorden moeten vervangen kunnen worden vanuit een lijst met suggesties gebaseerd op de spraakinvoer	Could	❌

6.7 Conclusie

Het prototype voldoet aan alle ‘must have’ en ‘should have’ functionaliteiten. Twee ‘could have’ requirements vielen buiten de planning. Overigens is het prototype volledig werkend doordat alle ‘must have’ en ‘should have’ functionaliteiten zijn gerealiseerd.

Het trainen van het smalle Whisper model op de Nederlandse versie van de Common Voice 11 dataset resulteert in een WER van 11.6177. Het trainen op een eigen dataset gebaseerd op een GGZ behandelplan resulteert in een WER van 9.82.

Er zijn geen optimalisaties nodig aangezien de WER binnen de range van 5-10% valt wat wordt beschouwd als goede kwaliteit en klaar voor gebruik.

7. Fase 5: Validatie

In de validatiefase wordt getest of de applicatie het probleem daadwerkelijk oplost. Dit gebeurt door de tijd te meten van invoer via spraak en typen door iemand die een bepaalde tekst nog niet eerder heeft gezien. Als de invoer via spraak sneller/makkelijker is dan typen, is de applicatie klaar om geïntegreerd te worden met de huidige software.

7.1 Validatie

Voor het valideren van het prototype zijn 10 zinnen samengesteld bestaande uit 5 korte en 5 langere zinnen. Deze zinnen zijn gebaseerd op de dataset voor Whisper Fine-Tuning, maar zijn door elkaar gehusseld om spraakinvoer geen voordeel te geven. Gedurende de invoer wordt gemeten hoe lang de invoer duurt, hoeveel fouten er gemaakt worden en hoe lang het gemiddeld duurt om een foutief woord te corrigeren. De invoer via typen is uitgevoerd op een laptop aangezien dit het meest gebruikte apparaat is voor het schrijven van rapportages volgens de enquêteresultaten.

Tekst	**Invoertijd typen (aantal fouten)**	**Invoertijd spraak (aantal fouten)**	**Verschil (%)**
1. De client is binnen Mediant opgenomen op een afdeling.	7s (0 fouten)	4s (0 fouten)	57%
2. De cliënt heeft met het Cimot een gesprek gehad.	7s (0 fouten)	3s (0 fouten)	43%
3. De cliënt staat nu bij de RIBW op de wachtlijst.	9s (0 fouten)	5s (0 fouten)	56%
4. Cliënt is op de afdeling WBT Hengelo opgenomen.	8s (0 fouten)	4s (0 fouten)	50%
5. De cliënt werd opgenomen met een RM.	7s (0 fouten)	4s (0 fouten)	57%
6. Een psychotische stoornis in het kader van een schizofrenie is de reden van behandeling.	14s (1 fout)	6s (1 fout)	43%
7. Vanwege het dealen in drugs aan mede-cliënten heeft cliënt een aantal keer een time-out gekregen in de afgelopen maanden.	18s (2 fouten)	7s (1 fout)	39%
8. Inmiddels heeft de patiënt het gebruik van depotmedicatie geaccepteerd nadat Dhr. onder curatele is geplaatst.	19s (0 fouten)	7s (1 fout)	37%
9. Vermoedelijk wordt regelmatig geconstateerd dat de cliënt verslavende middelen gebruikt en ook dat hij probeert lotgenoten daarbij te betrekken.	25s (3 fouten)	8s (1 fout)	32%
10. In de geestelijke gezondheidszorg worden regelmatig metingen gedaan waarbij de toestand van de cliënten met het oog op evaluatie en eventueel bijsturing van de behandeling gemeten wordt. Dit heet Routine outcome monitoring en wordt afgekort als 'ROM'.	40s (1 fout)	15s (1 fout)	38%

7.2 Conclusie

De validatie laat duidelijk zien dat tijdens het typen er meer fouten gemaakt worden, dit komt bij het typen vooral voor bij langere zinnen waarbij er meerdere langere en/of complexere woorden in een zin zitten. De invoer door middel van spraakherkenning daarentegen is in de gevallen van deze zinnen gemiddeld 45% sneller dan typen. Het kost een gebruiker gemiddeld 2,5 seconde om een foutief woord te corrigeren (Attendi, z.d.).

8. Fase 6: Integratie

In de integratiefase is het tijd om het product te introduceren aan de gebruikers van CARE4 en USER. Dit bestaat uit het integreren van het prototype met de Avinty applicaties. Doordat deze fase buiten de scope van het project valt zal in deze fase een advies opgesteld worden hoe het softwaresysteem het beste geïntegreerd kan worden met de huidige software. Daarnaast wordt onderzocht welke uitdagingen er komen kijken bij het uitrollen van spraakherkenningstechnologie in zorgsoftware.

8.1 Uitdagingen

Er zijn specifieke uitdagingen die komen kijken bij het uitrollen van spraakherkenningstechnologie in zorgsoftware, hieronder een aantal voorbeelden:

- **Schaling:** Spraakherkenningstechnologie moet vaak in staat zijn om te werken met grote hoeveelheden spraakgegevens. Dit kan een uitdaging zijn bij de deployment, aangezien de technologie veel rekenkracht en opslagcapaciteit vereist (MATLAB & Simulink, z.d.).

- **Compliance:** Zorg software moet voldoen aan de wettelijke eisen voor gezondheidszorg zoals de GDPR in Europa. Dit kan een uitdaging zijn bij de deployment, aangezien de technologie vaak gevoelige patiëntgegevens verwerkt. Deze applicatie zal zelf in-house gehost moeten worden, of bij een hostingpartij waarbij de juiste ISO en NEN certificeringen aanwezig zijn waaronder de ISO 27001 en NEN 7510. Deze norm geeft aan dat de organisatie vertrouwelijk en integer omgaat met de patiëntgegevens (Certificering-keuring.nl, z.d.).

8.2 Benodigdheden

Het uitrollen van spraakherkenningstechnologie kan een significante investering zijn, zowel in termen van financiële middelen als menselijke middelen. Er zijn verschillende onderdelen die nog uitgevoerd moeten worden voordat het prototype geïmplementeerd kan worden met de huidige software van Avinty.

Flutter package publiceren

Publish naar pub.dev: Flutter packages kunnen gepubliceerd worden naar pub.dev, de officiële package repository van Flutter. Hierdoor zijn de packages toegankelijk voor iedereen die de Flutter-ontwikkelomgeving gebruikt (Flutter, z.d.).

Server

Voor de server is een hosting platform vereist waarbij het mogelijk is om Python apps te bouwen, te implementeren en deze te monitoren. Daarbij is het handig om het prototype te hosten op een server die automatisch het aantal rekenkracht en opslagcapaciteit kan schalen wanneer nodig. Het automatisch kunnen schalen van rekenkracht en opslagcapaciteit is één van de grote voordelen van het hosten bij een hostingpartij in vergelijking met het in-house hosten van de applicatie. Dit zou zelf ook mogelijk zijn, maar hiervoor is de aanschaf van dure servers vereist. Google Cloud is een hosting provider die beschikt over de juiste certificaten en waarbij automatisch schalen ook mogelijk is (Google Cloud, z.d.).

9. Fase 7: Verbeteren

In de verbeteringsfase wordt onderzocht welke features ingezet kunnen worden om het prototype en daarbij de gebruikerservaring te verbeteren na het uitrollen. Doordat deze fase buiten de scope van het project valt zal in deze fase een advies opgesteld worden over hoe het prototype verbeterd kan worden.

In deze paragraaf wordt vooral ingegaan hoe het prototype zelf qua functionaliteit verbeterd kan worden. In het volgende hoofdstuk ‘Discussie & Conclusies’ zal verder ingegaan worden over hoe het spraakherkenning model zelf verbeterd kan worden vanwege huidige beperkingen, ofwel limitaties.

9.1 Nice to have functionaliteiten

In de testfase is geconcludeerd dat aan alle ‘must have’ en ‘should have’ functionaliteiten uit de lijst met functionele requirements is voldaan. Dit houdt ook direct in dat niet voldaan is aan de ‘could have’ functionaliteiten. Daarom worden deze functionaliteiten in deze fase verder toegelicht.

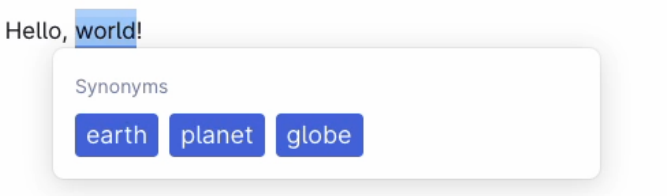
9.1.1 Tekst vervangen via spraak

Door gebruikers de mogelijkheid aan te bieden om tekst te vervangen door de spreken kunnen gebruikers vloeiend schakelen tussen invoer via stem (spraak) en aanraking (typen). Het vervangen van tekst via spraak zou gedaan kunnen worden door het selecteren van een stuk tekst welke vervolgens vervangen kan worden door te spreken. Doordat de invoer zowel via spraak en aanraking mogelijk zou kunnen zijn, hebben gebruikers de mogelijkheid om bijvoorbeeld een rapport af te maken door tekst te typen.

9.1.2 Woorden vervangen door suggesties

In het geval dat het model een woord toch nog verkeerd heeft herkend zou de mogelijkheid om een woord snel te kunnen vervangen vanuit een lijst met suggesties gebaseerd op de spraakinvoer tijd kunnen besparen als alternatief voor handmatig foutieve woorden corrigeren.

Ter verduidelijking, de lijst met suggesties verschijnt in de vorm van een kleine modal zodra er op een woord geklikt wordt. Zodra een woord vervangen is door een suggestie wordt deze feedback teruggestuurd worden naar het spraakherkenning model. Het model kan op deze manier leren van de feedback en zichzelf blijven verbeteren.



Illustratie ter verduidelijking, synoniemen zijn in dit geval suggesties gebaseerd op de spraakinvoer.

10. Conclusie

In dit onderzoek is gezocht naar een antwoord op de vraag: ‘Hoe kan spraakherkenningstechnologie geoptimaliseerd worden in zorgsoftware zodat het rapporteren makkelijker gemaakt kan worden voor zorgprofessionals?’. Hiervoor is een onderzoek uitgevoerd naar hoe spraakherkenningstechnologie veelgebruikte afkortingen, medische termen en jargon kan herkennen.

Zorgprofessionals ervaren stress als er tussen bezoeken geen tijd is om belangrijke informatie te noteren. Later op de dag rapporten uitwerken zonder aantekeningen kost veel tijd en resulteert in dat belangrijke details verloren gaan. Via de smartphone een rapportage typen is onhandig door het kleine formaat.

Uit de onderzoeksresultaten is gebleken dat bestaande services geen oplossing zijn omdat de maandelijkse kosten te hoog zijn bij bedrijven die beschikken over de juiste certificeringen en een spraakherkenning model getraind voor medische doeleinden in het geval van 27.000 gebruikers (USER). Overige services beschikken of niet over de juiste certificeringen of bieden geen model aan die specifiek getraind is voor medische doeleinden.

Door middel van een dataset bestaande uit audio en bijhorende transcripts kan een bestaand spraakherkenning model die al een goede basis heeft van de Nederlandse taal getraind worden d.m.v. Machine Learning om nieuwe woorden te herkennen.

Uit dit onderzoek is gebleken dat een spraakherkenning model die getraind is voor medische doeleinden gemiddeld een tijdsbesparing van 45% kan opleveren. Door kort na het contactmoment de informatie in te spreken en een rapport als concept op te slaan gaan belangrijke details nauwelijks verloren. Deze oplossing zorgt ook voor een toename in werkplezier, minder stress, verbeterde zorg voor cliënten en completere dossiers. Dit draagt direct bij aan de kwaliteit van zorg.

Overigens is het medische model getraind op een beperkte dataset waardoor de nauwkeurigheid niet het best mogelijke resultaat is. Dit komt omdat een kleine dataset niet voldoende representatief is voor de diversiteit aan spraak die het model in de praktijk zal tegenkomen, waardoor het model minder in staat is om spraak te herkennen die afwijkt van de spraak die het tijdens het trainen gehoord heeft.

11. Discussie

Taalbarrières
Spraakherkenningstechnologie werkt het best als de gebruiker een duidelijke en standaardtaal spreekt. Doordat het model getraind is op een beperkte dataset zal de dataset niet voldoende representatief zijn voor de diversiteit aan spraak die het model in de praktijk zal tegenkomen. Hierdoor is het model minder in staat om spraak te herkennen die afwijkt van de spraak die het model tijdens de training gehoord heeft.

Achtergrondgeluiden
Als zorgprofessionals spraak-naar-tekst onderweg gebruiken kan er mogelijk sprake zijn van veel omgevingsgeluid. De nauwkeurigheid van het spraak-naar-tekst model neemt af bij te veel omgevingsgeluid. In het prototype wordt momenteel geen rekening gehouden met achtergrondgeluiden.

Gebruiksvriendelijkheid
Zorgprofessionals hebben vaak weinig tijd waardoor de spraakherkenningstechnologie eenvoudig te gebruiken moet zijn voor zorgprofessionals, inclusief degenen die minder vaardig zijn met technologie. Dit kan een uitdaging zijn bij de deployment, aangezien de technologie geavanceerde functies heeft die mogelijk moeilijk te begrijpen zijn voor sommige gebruikers. Doordat de focus in het geval van het prototype vooral bij de backend lag is er vrijwel geen tijd besteed aan de gebruikersinterface.

Privacy en veiligheid
Het uitspreken van een rapportage brengt privacy risico’s met zich mee. Als een zorgprofessional niet goed oplet, kunnen mensen ongewenst meeluisteren. Deze audio bevat vaak gevoelige informatie van cliënten.

Schrijffouten
Soms bevat het resultaat van de spraak-naar-tekst een schrijffout. Het is niet de bedoeling dat er fouten in de dossiers komen.

12. Aanbevelingen

Het onderzoek concludeert dat een medisch spraakherkenning model gemiddeld voor een tijdsbesparing van 45% kan zorgen wat bijdraagt aan een toename in werkplezier, minder stress voor de zorgprofessional, completere dossiers en verbeterde zorg voor cliënten. Hierdoor is de aanbeveling om het prototype en het spraakherkenning model door te ontwikkelen makkelijk te maken.

Overigens wordt in het huidige prototype en spraakherkenning model niet overal rekening mee gehouden. Hieronder zijn aanbevelingen te vinden die dieper ingaan op de ‘Discussie’ uit het vorige hoofdstuk waarbij beantwoord wordt hoe de huidige limitaties opgelost kunnen worden.

Taalbarrières
Doordat het model getraind is op een beperkte dataset zal de dataset niet voldoende representatief zijn voor de diversiteit aan spraak die het model in de praktijk zal tegenkomen. Om deze limitatie op te lossen zal het model getraind moeten worden op ten minste een paar honderd uur aan audio bestaande uit veel verschillende sprekers, waaronder sprekers met accenten.

Nieuwe binnenkomende audio data uit de praktijk (die de server automatisch opslaat als wav bestand) kan ook gebruikt worden om het model nauwkeuriger te maken. In de praktijk bekijkt de zorgprofessional het resultaat van de spraak-naar-tekst en corrigeert deze waar nodig. De audio en transcripten kunnen gebruikt worden als trainingsmateriaal voor het model. Door iteratie wordt de nauwkeurigheid van het model constant verbeterd.

Achtergrondgeluid
De nauwkeurigheid van de spraak-naar-tekst neemt af bij teveel omgevingsgeluid. Geef gebruikers visuele feedback over de kwaliteit van de opname, zo kan de gebruiker gaandeweg steeds beter leren wanneer de technologie optimaal werkt.

Gebruiksvriendelijkheid
Zorgprofessionals hebben vaak weinig tijd waardoor de applicatie eenvoudig te gebruiken en te begrijpen moet zijn. Door samen met zorgprofessionals een design te ontwerpen en deze door middel van iteraties te testen en te verbeteren kan er een gebruikersinterface ontwikkeld worden die goed aansluit het werkproces van een zorgprofessional.

Privacy en veiligheid
Het uitspreken van een rapportage brengt privacy risico’s met zich mee. Informeer gebruikers over de risico’s van de spraakherkenning technologie en hoe daar goed mee omgegaan kan worden. Opnemen in een rustige omgeving is niet alleen goed voor de kwaliteit van spraak- naar-tekst, maar ook een stuk veiliger.

Schrijffouten
Doordat het resultaat van de spraak-naar-tekst soms schrijf fouten kan bevatten is het belangrijk dat een ingesproken bericht dat nog niet aangepast is duidelijk gemarkeerd wordt als een concept rapportage.

Bibliografie

Amazon Transcribe Medical. (z.d.). Amazon Web Services, Inc. Geraadpleegd op 30 augustus 2022, van https://aws.amazon.com/transcribe/medical/

Attendi. (z.d.). Geraadpleegd op 29 september 2022, van https://www.attendi.nl/speech-service/

Attendi. (z.d.). Attendi Speech Service Vier maanden in de praktijk bij Kwintes (GGZ). https://docplayer.nl/227504476-Attendi-speech-service-vier-maanden-in-de-praktijk-bij-kwintes-ggz.html

Avinty. (2021, 23 november). CARE4 Het Dossier. Geraadpleegd op 13 september 2022, van https://avinty.com/zorgoplossingen/care4-het-dossier/

Avinty. (2022, 26 augustus). Vernieuwing in de GGZ. Geraadpleegd op 13 september 2022, van https://avinty.com/ggz/

Baarda, B. (2019). Dit is onderzoek! : handleiding voor kwantitatief en kwalitatief onderzoek (3de editie). Verkregen van: https://search.saxionbibliotheek.nl/plugin/Koha/Plugin/EDS/opac/eds-detail.pl?q=Retrieve?an=2205619|dbid=nlebk&resultid=3

Benders, L. (2022, 17 oktober). Enquêtes correct gebruiken in je scriptie. Scribbr. https://www.scribbr.nl/onderzoeksmethoden/enquete-in-je-scriptie/

Byrne, J. (z.d.). What are the 7 stages of a new product development process? Geraadpleegd op 8 september 2022, van https://www.cognidox.com/blog/7-stages-of-new-product-development-process

Certificering-keuring.nl. (z.d.). Wat is het verschil tussen de NEN 7510 en de ISO 27001?, van https://www.certificering-keuring.nl/wat-is-het-verschil-tussen-de-nen7510-en-de-iso27001

Clean architecture. (2019, 1 maart). WhatIs.Com. Geraadpleegd op 23 december 2022, van https://whatis.techtarget.com/definition/clean-architecture

Fine-Tune Whisper For Multilingual ASR with Transformers. (z.d.). https://huggingface.co/blog/fine-tune-whisper

Flutter. (z.d.). Developing packages & plugins. https://docs.flutter.dev/development/packages-and-plugins/developing-packages

Flutter. (z.d.). Multi-Platform. https://flutter.dev/multi-platform

Foster, K. (2022, 4 oktober). The Top Free Speech-to-Text APIs, AI Models, and Open Source Engines. News, Tutorials, AI Research. https://www.assemblyai.com/blog/the-top-free-speech-to-text-apis-and-open-source-engines/

Frey, D. D., Herder, P. M., Wijnia, Y., Subrahmanian, E., Katsikopoulos, K., & Clausing, D. P. (2008). The Pugh Controlled Convergence method: model-based evaluation and implications for design theory. Research in Engineering Design, 20(1), 41–58. https://doi.org/10.1007/s00163-008-0056-z

Google Cloud. (z.d). - NEN (Netherlands) - Compliance. https://cloud.google.com/security/compliance/nen-netherlands

Google Play. (z.d.). Gboard - the Google Keyboard. Geraadpleegd op 30 augustus 2022, van https://play.google.com/store/apps/details?id=com.google.android.inputmethod.latin

Heck, P. (2020, 20 november). ICT Research Methods for Machine Learning Engineering. Fontys. https://fontysblogt.nl/ict-research-methods-for-machine-learning-engineering/

Hoftijzer, M., & Korte, P. (2013). Onderneem! CE Ondernemerschap Theorieboek. Verkregen van: http://hoadd.noordhoff.nl/sites/7895/_assets/7895d15.pdf

Jenčo, M., & Černák, I. (2019). Application of a confrontation matrix in project teams quality management. Quality Access to success, 20(170), 73–77. Retrieved from: https://www.researchgate.net/profile/Michal-Jenco/publication/333249288_Application_of_a_confrontation_matrix_in_project_teams_quality_management/links/5e7de83d458515efa0adc514/Application-of-a-confrontation-matrix-in-project-teams-quality-management.pdf

Joshi, A., Kale, S., Chandel, S., & Pal, D. (2015). Likert Scale: Explored and Explained. Verkregen van: https://doi.org/10.9734/bjast/2015/14975

Kannappa, M. (2021, 12 december). Data Streaming via GRPC vs MQTT vs Websockets - Meiyappan Kannappa. Medium. https://msmechatronics.medium.com/data-streaming-via-grpc-vs-mqtt-vs-5c30dd205193

Kumar, A. (2021, 13 juni). Machine Learning - Training, Validation & Test Data Set. Data Analytics. https://vitalflux.com/machine-learning-training-validation-test-data-set/

MATLAB & Simulink (z.d.). Supervised Learning. Geraadpleegd op 10 januari 2023, van https://nl.mathworks.com/discovery/supervised-learning.html?s_tid=srchtitle_+Supervised+learning_1

MATLAB & Simulink (z.d.). What is Deep Learning? | How It Works, Techniques & Applications. Geraadpleegd op 23 september 2022, van https://nl.mathworks.com/discovery/deep-learning.html

MATLAB & Simulink (z.d.). What is Machine Learning? | How it Works, Tutorials, and Examples. Geraadpleegd op 23 september 2022, van https://nl.mathworks.com/discovery/machine-learning.html

MATLAB & Simulink (z.d.). What is a Neural Network? | How it Works, Tutorials, and Examples. Geraadpleegd op 23 september 2022, van https://nl.mathworks.com/discovery/neural-network.html

Medium. (2021, 7 december). Speech recognition is hard — Part 1 - Towards Data Science. Medium. https://towardsdatascience.com/speech-recognition-is-hard-part-1-258e813b6eb7

Microsoft. (2022, 30 november). Test accuracy of a Custom Speech model - Speech service - Azure Cognitive Services. Microsoft Learn. Geraadpleegd op 7 januari 2022, van https://learn.microsoft.com/en-us/azure/cognitive-services/speech-service/how-to-custom-speech-evaluate-data?pivots=speech-studio

Mozilla Common Voice. (z.d.). https://commonvoice.mozilla.org/en/languages

O'Connor, R. (2022, 13 december). How to Run OpenAI's Whisper Speech Recognition Model. News, Tutorials, AI Research. https://www.assemblyai.com/blog/how-to-run-openais-whisper-speech-recognition-model/

Onderzoekdoen.nl. (2017, 1 juli). 5-punts en 7-punts likertschaal, waarom kiezen voor welke likert schaal? Verkregen van: https://www.onderzoekdoen.nl/enquete-onderzoek/likert-schaal/

Quora. (z.d.). How many voice data is needed for the training of commercial DNN-HMM ASR system with accuracy of 90%? Verkregen van: https://www.quora.com/How-many-voice-data-is-needed-for-the-training-of-commercial-DNN-HMM-ASR-system-with-accuracy-of-90

Radford, A. (2022, 6 december). Robust Speech Recognition via Large-Scale Weak Supervision. arXiv.org. https://arxiv.org/abs/2212.04356

Revilla, M., & Ochoa, C. (2017). Ideal and maximum length for a web survey. Ideal and maximum length for a web survey. Verkregen van: https://doi.org/10.2501/IJMR-2017-039

Röpke, W (2019). Training a Speech-to-Text Model for Dutch on the Corpus Gesproken Nederlands. https://ceur-ws.org/Vol-2491/paper60.pdf

Scriptix. (2021, 14 oktober). Everything about speech to text ➔ Software & API ➔ Scriptix. Scriptix - Speech recognition engines built for you. https://www.scriptix.io/speech-to-text/

Spraakherkenning GGZ. (2022, 29 augustus). Cedere. Geraadpleegd op 29 september 2022, van https://www.cedere.nl/medici/spraakherkenning-ggz/

SurveyMonkey (z.d.). Het aantal respondenten dat u nodig hebt berekenen. Geraadpleegd op 11 oktober 2022, van https://help.surveymonkey.com/nl/solutions/calculating-respondents/

Tejedor-García, C. (z.d.). Towards an Open-Source Dutch Speech Recognition System for the Healthcare Domain. ACL Anthology. https://aclanthology.org/2022.lrec-1.110/

Verhoeven, N. (2018). Wat is onderzoek? - Praktijkboek voor methoden en technieken (6de editie). Amsterdam, Nederland: Boom uitgevers Amsterdam.

Wingman. (z.d.). Understanding Word Error Rate (WER) in Automatic Speech Recognition (ASR). Geraadpleegd op 22 december 2022, van https://www.trywingman.com/blog-posts/what-is-word-error-rate-in-automatic-speech-recognition

Ycombinator. (z.d.). Facebook has released wav2letter++. https://news.ycombinator.com/item?id=22155344

Appendix

Appendix A: Enquêtevragen

Introductie

Hey! Ik ben Thomas, student aan de Hogeschool Saxion en voor mijn afstudeerstage bij Avinty ben ik onderzoek aan het doen hoe een softwareoplossing het rapporteren voor zorgprofessionals makkelijker kan maken.

Daarom ben ik op zoek naar mensen die in de zorg werken en te maken hebben met rapporteren.

Zou je mij willen helpen door anoniem een aantal vragen te beantwoorden. Het invullen van de enquête duurt ongeveer 3 minuten. De resultaten zullen uitsluitend voor deze opdracht gebruikt worden.

Alvast bedankt!

Categorie 1 - Algemene gegevens

Vraag: Wat je is geslacht? [Niet verplicht, mogelijkheid om vraag over te slaan]
Antwoordmogelijkheden: man, vrouw, wil ik niet zeggen

Vraag: Wat je leeftijd?
Antwoordmogelijkheden: 16-25, 26-30, 31-35, 36-39, 40-50, 50+

De volgende vragen/stellingen gaan over de realisatie van een spraak-naar-tekst software systeem specifiek ontwikkeld voor medische doeleinden die het invoeren van rapporten makkelijker maakt.

Het idee is dat direct na afloop van een cliënt bezoek alle informatie kan worden ingesproken. De ingesproken rapportages worden omgezet in tekst en verschijnen in het dossier als concept. Dit rapport kan op ieder moment van de dag in een korte tijd verwerkt worden tot een eindresultaat zonder dat hier een laptop bij nodig is.

Categorie 2 - Probleem

Vraag: Rapporteren is een tijdrovend en frustrerend onderdeel van het werk
Antwoordmogelijkheden: 7-punts Likertschaal (zeker oneens, oneens, een beetje oneens, neutraal, een beetje mee eens, mee eens, zeker mee eens)

Vraag: Ik ervaar rapporteren als lastig en omslachtig
Antwoordmogelijkheden: 7-punts Likertschaal (zeker oneens, oneens, een beetje oneens, neutraal, een beetje mee eens, mee eens, zeker mee eens)

Vraag: Ik maak tijdens afspraken notities in een notitieblok of in een notitie-app op mijn telefoon
Antwoordmogelijkheden: 7-punts Likertschaal (zeker oneens, oneens, een beetje oneens, neutraal, een beetje mee eens, mee eens, zeker mee eens)

Vraag: Ik ben na mijn afspraken vaak nog een uur bezig met het invoeren van rapporten
Antwoordmogelijkheden: 7-punts Likertschaal (zeker oneens, oneens, een beetje oneens, neutraal, een beetje mee eens, mee eens, zeker mee eens)

Vraag: Op welk moment na een afspraak verwerk jij het rapport?
Antwoordmogelijkheden:
- Direct na een afspraak
- Op een rustig moment tijdens de werkdag
- In de avonduren

<u>Categorie 3 - Oplossing</u>
Vraag: De optie om (concept) rapportages altijd en overal in te kunnen spreken met behulp van spraak-naar-tekst kan het rapporteren makkelijker maken
Antwoordmogelijkheden: 7-punts Likertschaal (zeker oneens, oneens, een beetje oneens, neutraal, een beetje mee eens, mee eens, zeker mee eens)

Vraag: Ik heb wel/geen ervaring met het inspreken van rapportages
Antwoordmogelijkheden:
- Ik heb WEL ervaring met het inspreken van rapportages
- Ik heb GEEN ervaring met het inspreken van rapportages

Vraag: Hoeveel tijd is er gemiddeld tussen afspraken om een concept rapport in te spreken?
Antwoordmogelijkheden:
- Daar is geen tijd voor
- Ongeveer 5 minuten
- Meer dan 5 minuten

Vraag: Ik denk dat als ik tijdens mijn werkdag, tussen afspraken door, eenvoudiger kan rapporteren door spraakinvoer dat ik mijn dag rustiger afsluit en meer tijd voor de cliënten heb
Antwoordmogelijkheden: 7-punts Likertschaal (zeker oneens, oneens, een beetje oneens, neutraal, een beetje mee eens, mee eens, zeker mee eens)

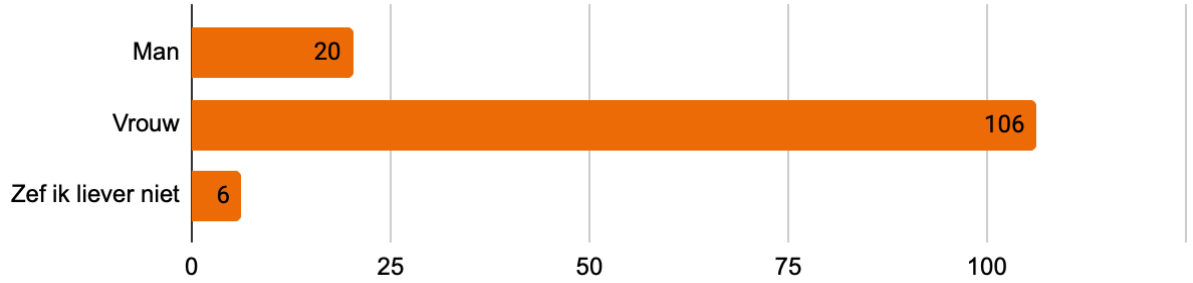
<u>Categorie 4 - Gedrag</u>
Vraag: Welk device gebruik jij tijdens het invoeren van rapporten?
Antwoordmogelijkheden: telefoon, tablet, laptop

Vraag: Heb je nog vragen, tips of opmerkingen?
Antwoordmogelijkheden: open antwoord

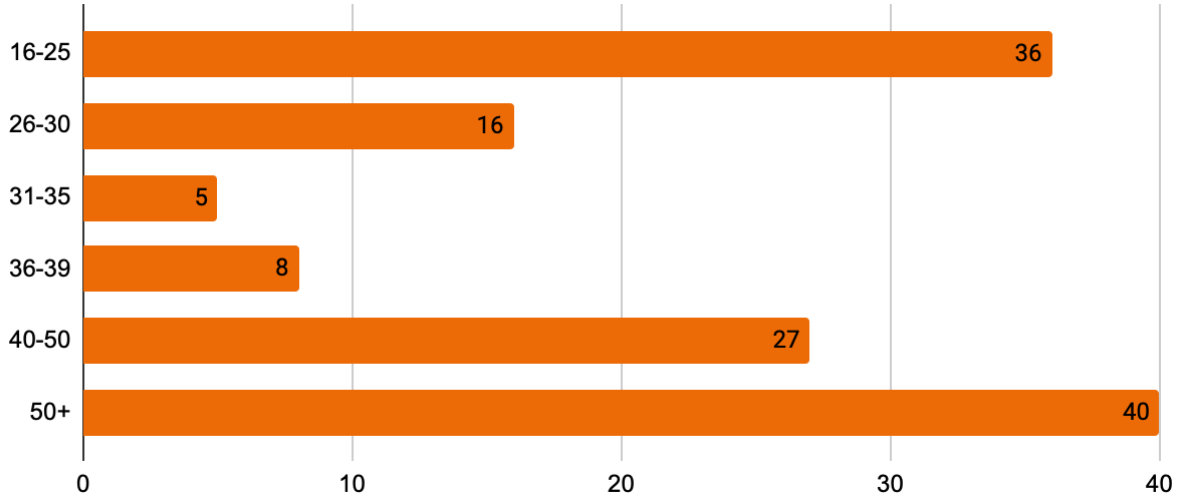
Appendix B: Enquêteresultaten

Categorie 1 - Algemene gegevens

Vraag: Wat je is geslacht?

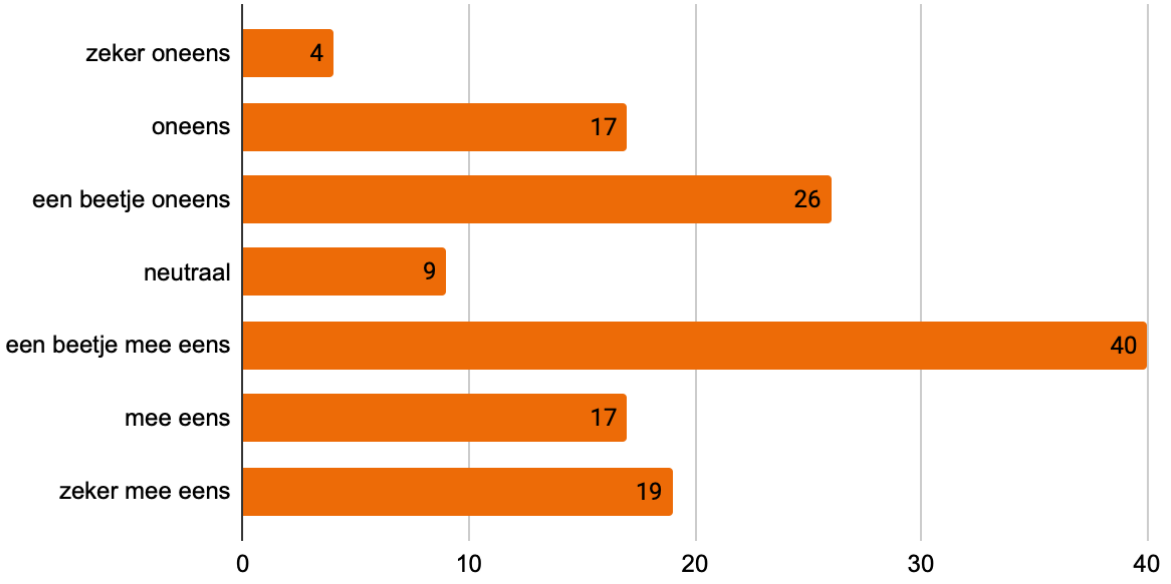


Vraag: Wat je leeftijd?

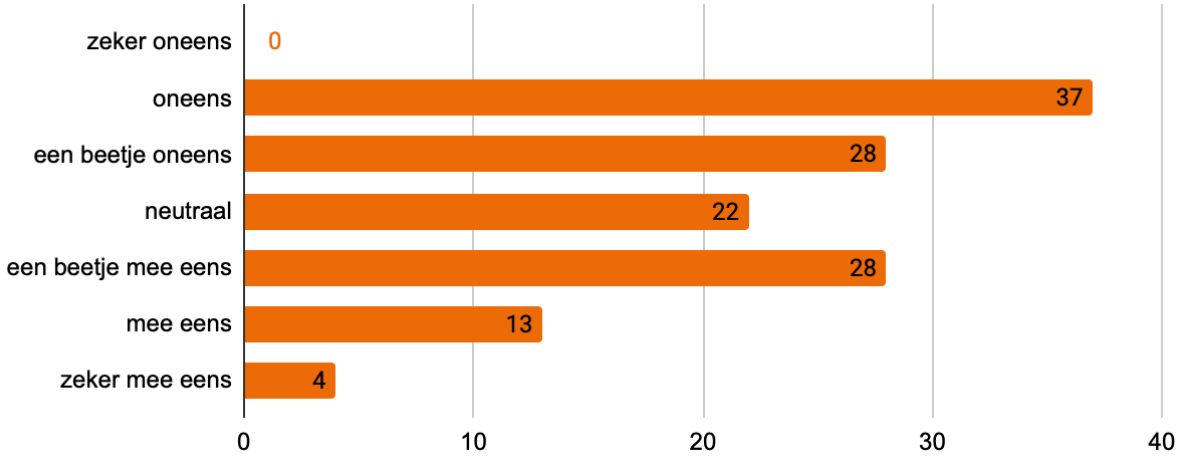


Categorie 2 - Probleem

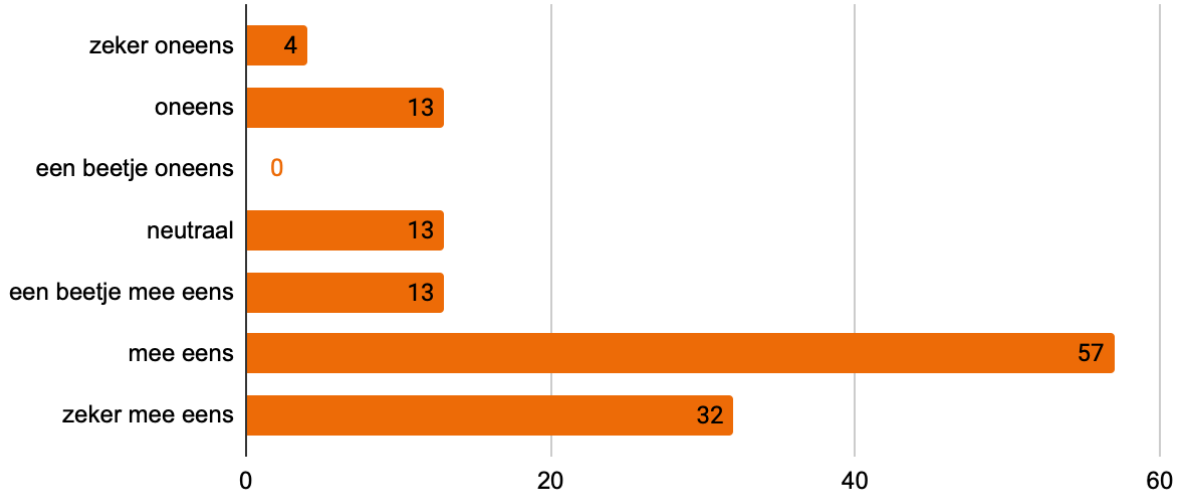
Vraag: Rapporteren is een tijdrovend en frustrerend onderdeel van het werk



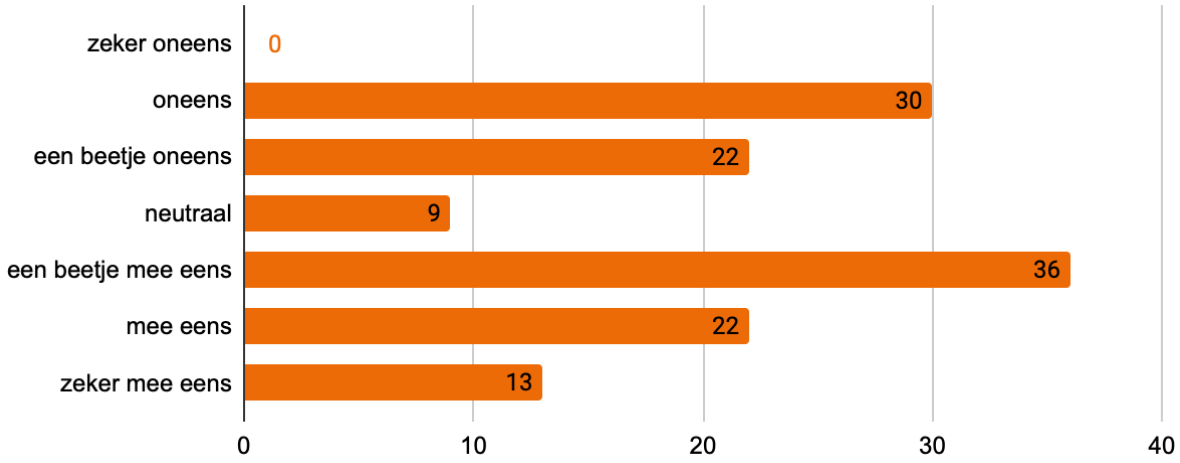
Vraag: Ik ervaar rapporteren als lastig en omslachtig



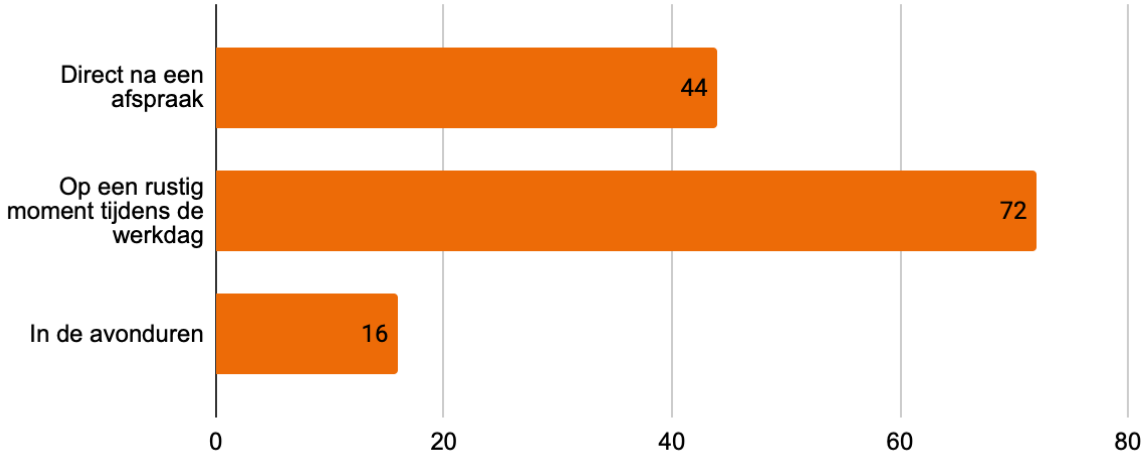
Vraag: Ik maak tijdens afspraken notities in een notitieblok of in een notitie-app op mijn telefoon



Vraag: Ik ben na mijn afspraken vaak nog een uur bezig met het invoeren van rapporten

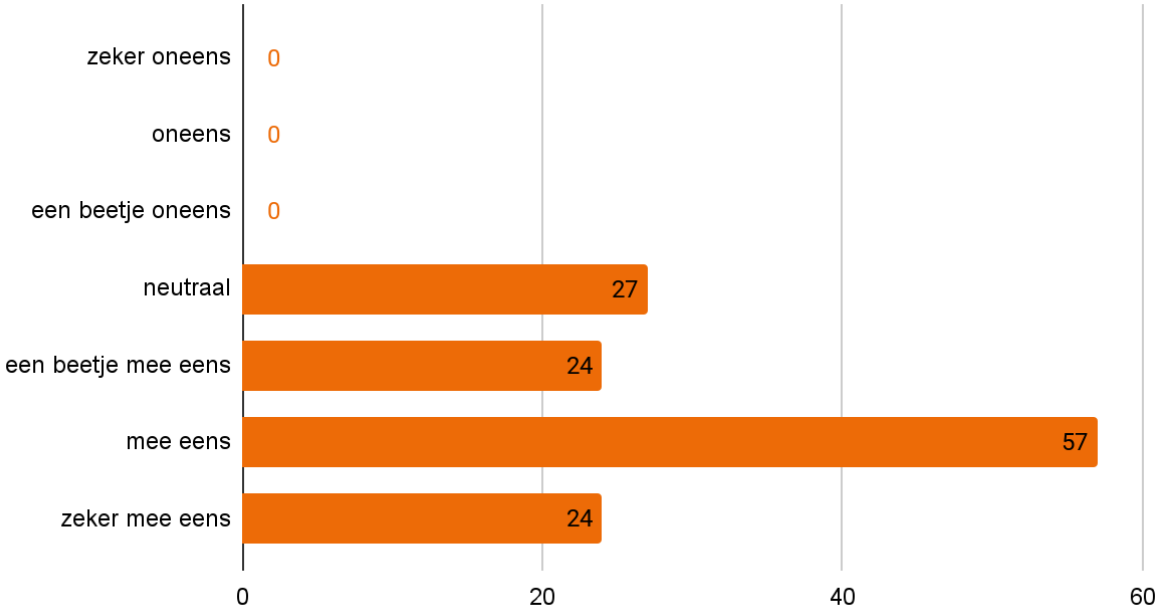


Vraag: Op welk moment na een afspraak verwerk jij het rapport?

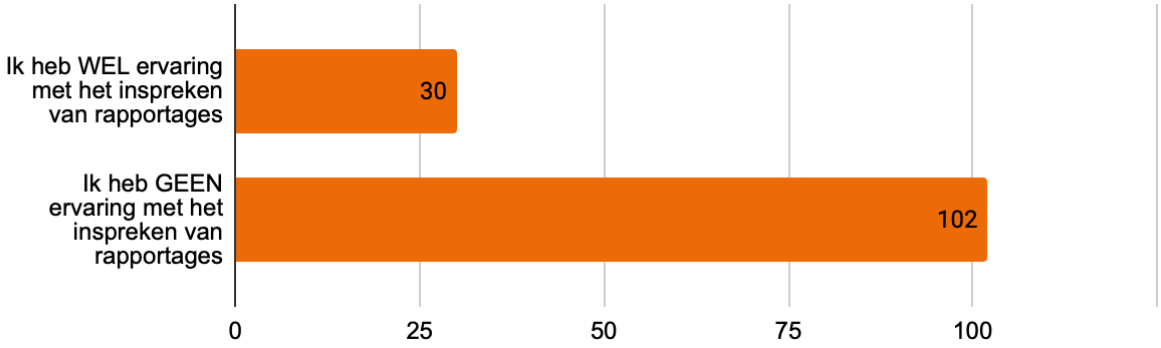


<u>Categorie 3 - Oplossing</u>

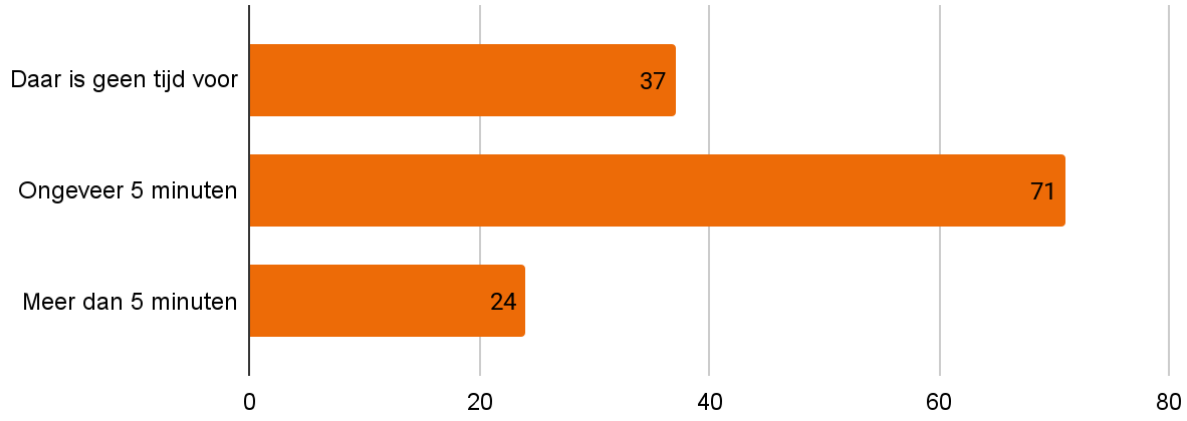
Vraag: De optie om (concept) rapportages altijd en overal in te kunnen spreken met behulp van spraak-naar-tekst kan het rapporteren makkelijker maken



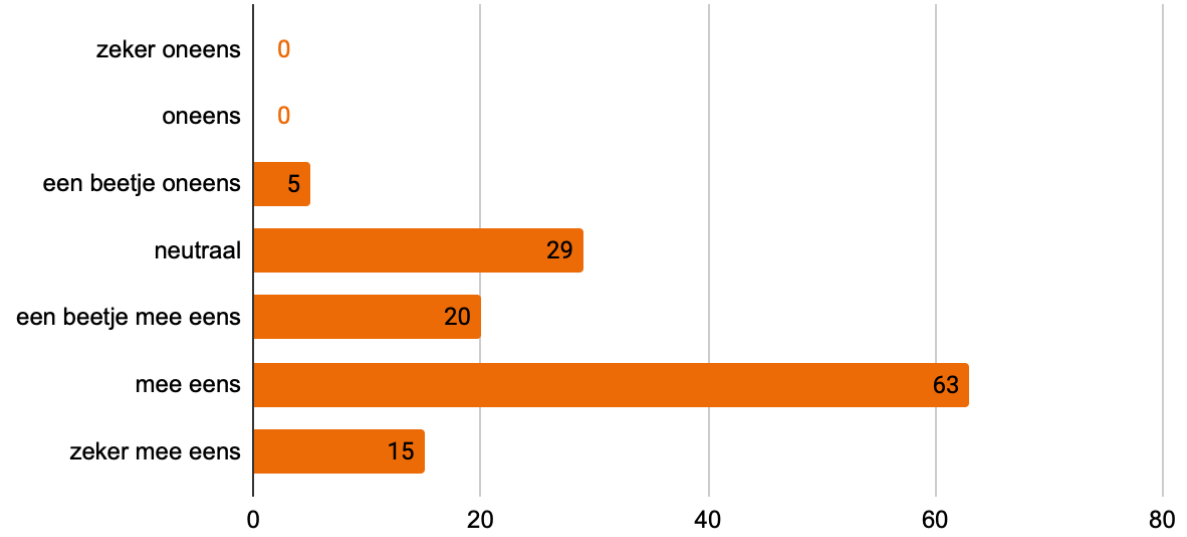
Vraag: Ik heb wel/geen ervaring met het inspreken van rapportages



Vraag: Hoeveel tijd is er gemiddeld tussen afspraken om een concept rapport in te spreken?

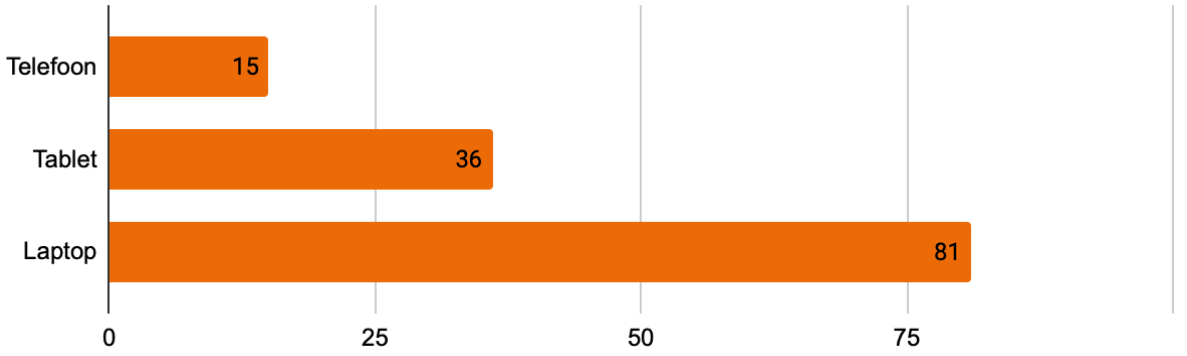


Vraag: Ik denk dat als ik tijdens mijn werkdag, tussen afspraken door, eenvoudiger kan rapporteren door spraakinvoer dat ik mijn dag rustiger afsluit en meer tijd voor de cliënten heb



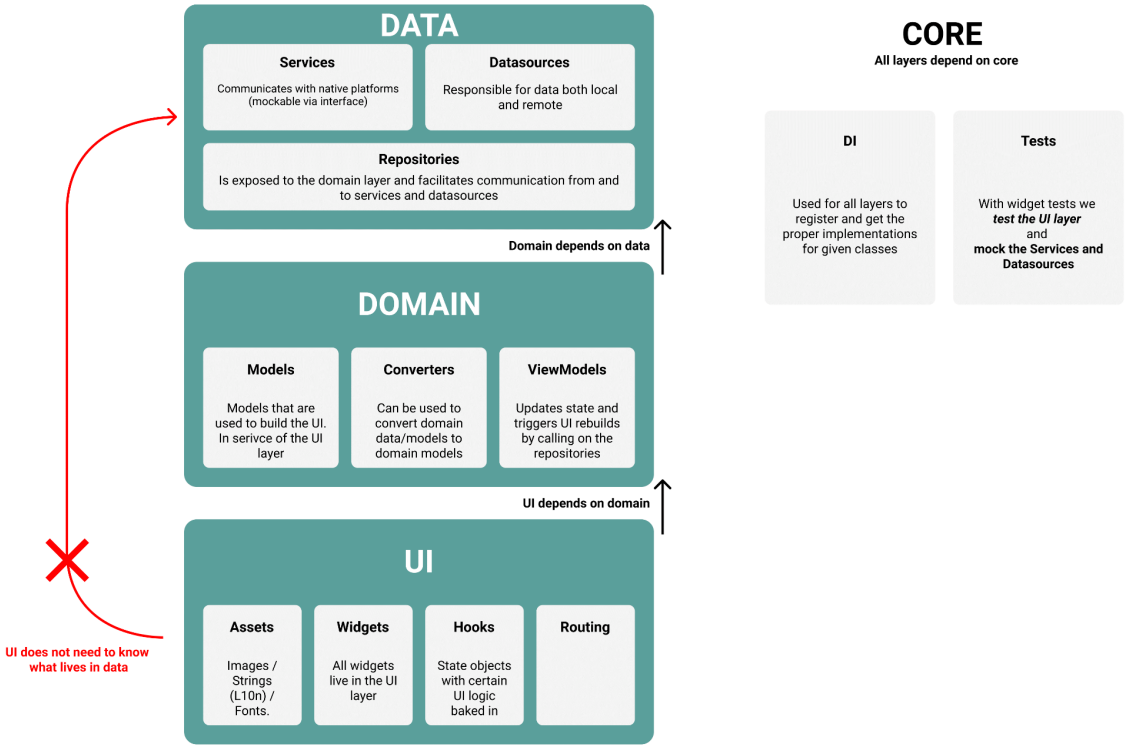
<u>Categorie 4 - Gedrag</u>

Vraag: Welk device gebruik jij tijdens het invoeren van rapporten?



Appendix C: Software architectuur

Een visuele representatie van het clean architecture model waarop de Flutter applicatie gebaseerd is.



Appendix D: Behandelplan GGZ

Man (1979) van Koreaanse origine door adoptie op 2,5 jarige leeftijd naar Nederland gekomen, werd met een IBS (brandgevaar) en aansluitend een RM (verwaarlozing) opgenomen binnen Mediant. Dhr. is bekend met schizofrenie, hij gebruikt daarbij middelen en stopte een aantal maal zijn medicatie. Dhr. ervaart dat hij volledig alleen op de wereld is en alles zelf op moet knappen. Dhr. is onder curatele geplaatst. Inmiddels heeft patiënt het gebruik van depotmedicatie geaccepteerd.

Cliënt is op 20-07-2017 opgenomen op afdeling WBT Hengelo. Cliënt gaat overwegend zijn eigen gang op de afdeling en verblijft meestentijds op zijn kamer. Dhr. is over het algemeen vriendelijk in contact. Van de verpleging ontvangt hij zijn voedingsgeld per week en hij bereidt zelfstandig zijn maaltijden.

Zo nu en dan maakt cliënt gebruik van de voorzieningen van de afdeling; kijkt hij TV in de huiskamer of rookt hij een sigaret op het balkon. Dhr. heeft eenmaal de afdelingsregel m.b.t. nuttigen van alcohol overtreden en was eenmaal te laat terug van een weekendverlof. Dhr. zegt voor 60% tevreden te zijn over de RM en zal afwachten wat de behandelaars van hem verwachten met betrekking tot begeleid wonen in de toekomst. Dhr. is van plan om bij de behandelplanbespreking aanwezig te zijn.

Reden tot behandeling:
Psychotische stoornis in het kader van een schizofrenie. Client was voor opname langdurig psychotisch waardoor hij ernstige overlast veroorzaakte, lichamelijk verslechterde door onvoldoende voeding en gevaar ontstond voor maatschappelijk verval/teloorgang.

Beschrijvende diagnose:
Man van Koreaanse origine door adoptie op 2,5 jarige leeftijd naar Nederland gekomen, werd met een IBS (brandgevaar) en aansluitend een RM (verwaarlozing) opgenomen binnen Mediant.

Cliënt is bekend met schizofrenie, hij gebruikt daarbij middelen en stopte een aantal maal zijn medicatie. Cliënt ervaart dat hij volledig alleen op de wereld is en alles zelf op moet knappen. Cliënt is onder curatele geplaatst. Inmiddels heeft cliënt zich neergelegd bij het gebruik van depotmedicatie geaccepteerd.

Bevindingen afgelopen periode:
Cliënt was in de afgelopen periode (6 mnd) overwegend rustig en vriendelijk aanwezig.

Op het moment is cliënt stabiel en tevreden over zijn huidige situatie.

Cliënt is overdag meer actief. Cliënt krijgt wekelijks een voedingsbudget waar hij zijn eigen maaltijden van bereid. Cliënt is er op aangesproken gezond voedsel aan te schaffen voor het voedingsbudget.

Cliënt krijgt maandelijks zijn depotmedicatie.

In de afgelopen maanden heeft cliënt een aantal keer een time-out gekregen vanwege dealen in drugs aan mede-cliënten.Tevens heeft cliënt een afdeling verbod gekregen voor afdeling BT en afdeling MZ

Regelmatig wordt geconstateerd of vermoed dat cliënt verslavende middelen gebruikt en ook dat hij probeert lotgenoten daarbij te betrekken. Door hun middelen aan te bieden, vermoedelijk met de bedoeling om daar geld mee te verdienen.

Cliënt heeft een gesprek gehad met het Cimot: hij staat nu op de wachtlijst van het RIBW. Een wachttijd van 1,5 tot 2 jaar is verwacht.

Appendix E: Dataset voor Whisper Fine-Tuning

1. Cliënt is opgenomen op afdeling WBT Hengelo.
2. Tevens heeft cliënt een afdelingsverbod gekregen voor afdeling BT.
3. Tevens heeft cliënt een afdelingsverbod gekregen voor afdeling MZ.
4. De cliënt heeft een gesprek gehad met het Cimot.
5. De cliënt staat nu op de wachtlijst van het RIBW.
6. Dhr. heeft eenmaal de afdelingsregel m.b.t. nuttigen van alcohol overtreden.
7. De cliënt is opgenomen binnen Mediant.
8. De cliënt werd met een IBS opgenomen.
9. De cliënt werd met een RM opgenomen.
10. De cliënt kijkt TV in de huiskamer of rookt een sigaret op het balkon.
11. Dhr. zegt voor 60% tevreden te zijn over de RM en zal afwachten wat de behandelaars van hem verwachten met betrekking tot begeleid wonen in de toekomst.
12. De reden van behandeling is een psychotische stoornis in het kader van een schizofrenie.
13. Dhr. is van plan om bij de behandelplanbespreking aanwezig te zijn.
14. Dhr. is onder curatele geplaatst. Inmiddels heeft de patiënt het gebruik van depotmedicatie geaccepteerd.
15. Routine outcome monitoring afgekort als 'ROM' is een methodiek in de geestelijke gezondheidszorg waarbij regelmatig metingen gedaan worden van de toestand van de cliënten met het oog op evaluatie en eventueel bijsturing van de behandeling.
16. Regelmatig wordt geconstateerd of vermoed dat de cliënt verslavende middelen gebruikt en ook dat hij probeert lotgenoten daarbij te betrekken.
17. De cliënt was voor opname langdurig psychotisch waardoor hij ernstige overlast veroorzaakte.
18. De cliënt krijgt wekelijks een voedingsbudget waar hij zijn eigen maaltijden van bereid.
19. In de afgelopen maanden heeft cliënt een aantal keer een time-out gekregen vanwege dealen in drugs aan mede-cliënten.
20. De cliënt verslechterde lichamelijk door onvoldoende voeding.

Appendix F: Test cases

Test	Omzetten van Nederlandse spraak naar tekst	TC 1 (Must)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'Goedemorgen, mijn naam is Thomas.'*	
Postconditie	De audio is succesvol getranscribeerd naar tekst. De tekst *'Goedemorgen, mijn naam is Thomas.'* staat in het invoerveld	
Status	GESLAAGD	

Test	Herkennen van medische termen, bedrijfsnamen	TC 2 (Must)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'De cliënt is opgenomen binnen Mediant.'*	
Postconditie	De audio is succesvol getranscribeerd naar tekst. De tekst *'De cliënt is opgenomen binnen Mediant.'* staat in het invoerveld	
Status	GESLAAGD	

Test	Audio verwerking van meerdere clients tegelijk	TC 3 (Must)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de clients	
Input	1. De eerste gebruiker drukt op de knop om het inspreken te starten 2. De tweede gebruiker drukt op de knop om het inspreken te starten 3. De eerste gebruikt spreekt *'De cliënt heeft een gesprek gehad met het Cimot.'* en de tweede gebruiker spreekt *'De cliënt staat nu op de wachtlijst van het RIBW.'*	
Postconditie	De audio van de clients is succesvol getranscribeerd naar tekst. De tekst *'De cliënt heeft een gesprek gehad met het Cimot.'* staat in het invoerveld voor de eerste gebruiker. De tekst *'De cliënt staat nu op de wachtlijst van het RIBW.'* staat in het invoerveld voor de tweede gebruiker	
Status	GESLAAGD	

Test	Pauzeren van spraakinvoer	TC 4 (Should)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'Dit is een test.'* 3. De gebruiker pauzeert de invoer 4. De gebruiker spreekt *'Dit is een test.'*	
Postconditie	De spraakinvoer is succesvol gepauzeerd. De tekst *'Dit is een test.'* staat in het invoerveld	
Status	GESLAAGD	

Test	Hervatten van spraakinvoer	TC 5 (Should)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'Dit is een test.'* 3. De gebruiker pauzeert de invoer 4. De gebruiker hervat de invoer 5. De gebruiker spreekt *'Dit is de tweede test'*.	
Postconditie	De spraakinvoer is succesvol hervat. De tekst *'Dit is een test. Dit is de tweede test.'* staat in het invoerveld	
Status	GESLAAGD	

Test	Opslaan van spraakinvoer als audiobestand	TC 6 (Should)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'Dit is een test.'*	
Postconditie	De spraakinvoer is succesvol opgeslagen als audiobestand (wav file) op de server. De tekst *'Dit is een test. Dit is de tweede test.'* staat in het invoerveld	
Status	GESLAAGD	

Test	Vervangen van tekst door selectie en spreken	TC 7 (Could)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'Dit is een test.'* 3. De gebruiker selecteert de tekst *'test'* *4.* De gebruiker spreekt *'prototype"*	
Postconditie	Het is niet mogelijk om tekst te vervangen door deze te selecteren en vervolgens te spreken. De tekst *'Dit is een test. Prototype.'* staat in het invoerveld	
Status	MISLUKT	

Test	Woorden vervangen uit lijst met suggesties	TC 8 (Could)
Actor	Server, Client	
Preconditie	De server is gestart en de invoerpagina wordt weergegeven op de client	
Input	1. De gebruiker drukt op de knop om het inspreken te starten 2. De gebruiker spreekt *'Dit is een test.'* 3. De gebruiker drukt op het woord 'test'	
Postconditie	Het is niet mogelijk om woorden te vervangen uit een lijst met suggesties. De tekst *'Dit is een test.'* staat in het invoerveld	
Status	MISLUKT	