Update: plan van aanpak generatieve AI opdracht 1.0

Wat is het project?

Het project is een onderzoek naar het gebruik van (generatieve) AI om de dienstverlening van SSC ICT te verbeteren door medewerkers te ondersteunen en interne processen te optimaliseren. In het bijzonder ligt de focus op generatieve large language models (LLM), omdat dit de grootste impact kan hebben op het dagelijks werk van werknemers van SSC-ICT.

De opdracht (Spraakbesturing & AI) is geaccordeerd door het SpB en loopt sinds begin 2023 (interne urencode **201202044.002**).

Waarom nu?

Generatieve AI wordt al door meer dan een kwart van de werknemers van de Rijksoverheid gebruikt. Het heeft de potentie om delen van zeer uiteenlopend werk te vergemakkelijken en kan veel tijd besparen. De keerzijde is dat nu diensten gebruikt worden (ChatGPT, BingChat, Google) die de data waarmee ze worden gevoed, zelf gebruiken. Door betaalde AI diensten te leveren of zelf diensten te ontwikkelen kunnen datastromen worden gecontroleerd en blijft het gebruik van AI veiliger. Bovendien kan generatieve AI worden toegespitst op domeinspecifieke taken wanneer het door SSC-ICT zelf wordt ingericht.

Wie gaat het gebruiken?

De beoogde gebruikers van de AI-toepassingen zijn alle medewerkers van SSC-ICT. De use case die voor dit project wordt uitgewerkt richt zich op de servicedesk. Het zou mogelijk zijn om een chatbot in te zetten voor de vervanging van het self-service portaal. De reden hiervoor is dat de toepassing duidelijk is, het risico laag is en er een goed afgebakende knowledge-base bestaat.

Hoe?

We onderzoeken de mogelijkheden voor het opzetten van een chatbot-assistent. Hierbij kijken we naar de mogelijkheid om zelf iets op te zetten op basis van beschikbare open source modellen en eventueel de mogelijkheid om deze verder te finetunen, maar ook naar kant en klare generatieve AI services in de Microsoft Azure omgeving. In de eindrapportage vergelijken we de resultaten en benoemen we relevante verschillen, ook als het aankomt op algemene voordelen of nadelen van open source ten opzichte van een Microsoft product.

Achtergrondinformatie

Het innovatieteam heeft tussen 2016 en 2022 uitgebreid onderzoek gedaan naar machine learning en AI. Hieruit werd de conclusie getrokken dat het hebben van een eigen AI implementatie technisch haalbaar en zeer wenselijk is. Verder zijn meerdere use cases voor deze technologie onderzocht en als geschikte kandidaat aangewezen waaronder chatbots, documentatie en interne lesstof generator. In 2023 heeft het service portfolio board het innovatie team gevraagd om een AI quickscan en impactanalyse te doen naar onder andere de inzet van generatieve AI. Dit project bouwt verder aan de eerder opgedane kennis.

De houding van de Rijksoverheid ten opzichte van AI toepassingen is nogal ambivalent. Aan de ene kant blijkt de vraag te bestaan, want veel werknemers maken voor hun werk al gebruik van verschillende (generatieve) AI diensten. Een goedgekeurd aanbod bestaat echter nog nauwelijks. Dit komt deels doordat de manier waarop de Rijksoverheid om moet gaan met AI verdeeldheid zaait. Een deel ziet beren op de weg en zou AI het liefst compleet buiten de deur houden. De toepassingen van AI blijven echter groeien en het zal binnen de toekomstige manier van werken onmogelijk zijn hier niet aan deel te nemen. Het is wel van belang dat bij het huidige project ook aandacht wordt besteed aan de impact van de AI en de ethische vraagstukken die dit met zich meebrengt.

Bovendien is het goed denkbaar dat er een politieke wens komt om als rijksoverheid meer met AI te doen. Zo komt in verschillende verkiezingsprogramma’s het thema AI terug en bestaat de mogelijkheid dat dit zich ook vertaalt naar het regeerakkoord. Het is daarom wenselijk dat onderzoek is gedaan naar de mogelijkheden op dit gebied.

Deelonderzoeken & Doelstellingen

1. Ontwikkeling open source Chatbot
   1. Open source LLM onderzoek

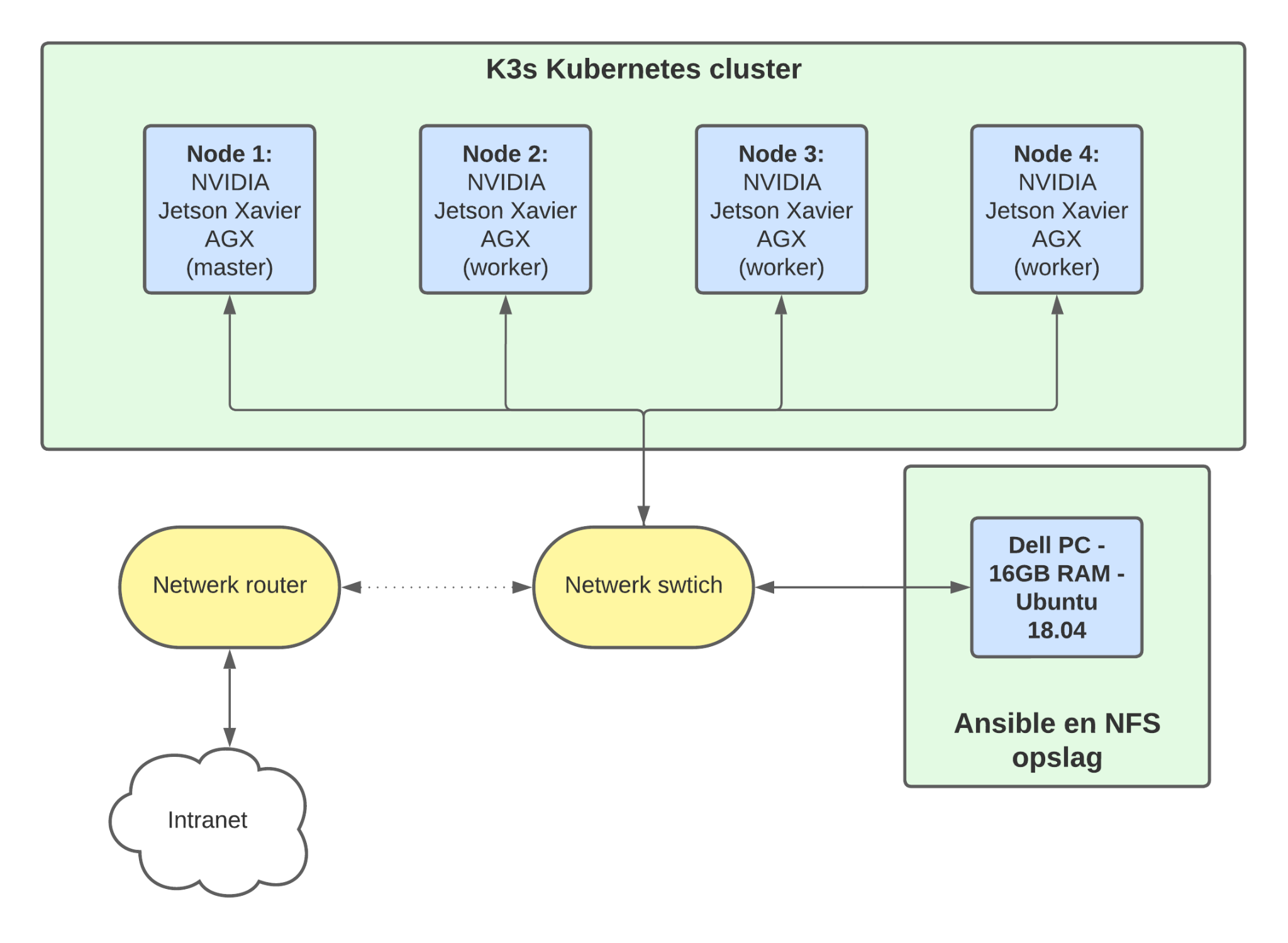
Er zijn veel open source large language base models beschikbaar. Deze modellen zijn veelal gratis volledig te downloaden in verschillende versies en groottes. Dit maakt het mogelijk om deze modellen lokaal te runnen en hiermee te experimenteren. In de eerste fase van dit project is een [onderzoek naar open source LLM’s](https://docs.google.com/document/d/1U_0FFPa-3OQnJ1r6bwaljcNCusrRCz4o/edit) uitgevoerd om uit te zoeken wat voor verschillende modellen bestaan en welke het meest geschikt zijn voor onze eigen toepassing.

* 1. Low-fidelity PoC

Hoewel het team beschikking heeft over 4 GPU’s, lukt het nog niet om hier modellen effectief op te draaien. Daarom wordt eerst een low fidelity proof of concept ontwikkeld waarbij een [quantized versie van LLAMA2](https://huggingface.co/TheBloke/Llama-2-7B-GGML)[[1]](#footnote-0) gedraaid wordt op een CPU. Hierbij wordt gebruikgemaakt van [LangChain](https://python.langchain.com/docs/get_started/introduction)[[2]](#footnote-1) om het model documenten van het self-service portaal van SSC-ICT mee te geven als knowledge base. Door te experimenteren met verschillende parameterinstellingen en prompts wordt de werking hiervan geoptimaliseerd. Er wordt ook een rudimentaire interface ontwikkeld waarin vragen gesteld kunnen worden.  
  
Deze fase resulteert in een basale versie van ons eigen open source model. Het zal nog niet optimaal functioneren, maar kan gebruikt worden om het idee achter dit project te valideren en om de eerste feedback te verzamelen.

* 1. Lokale chatbot

Zoals eerder genoemd hebben we beschikking over 4 GPU’s[[3]](#footnote-2). Deze zijn ook nodig om een volledig en goed functionerend LLM te draaien. Om dit effectief te doen, moeten de GPU’s aaneengeschakeld worden in een cluster (zie fig. 1). Zodra dit gerealiseerd is, kunnen de modellen die als de beste opties naar voren kwamen uit 1.1 worden getest. Dit testen kan in een lokale omgeving en zal worden toegespitst op de servicedesk use-case. Een verwachte uitkomst is dat modellen *finetuning* nodig hebben voordat ze echt goed functioneren. Dit kan echter veel tijd en resources kosten. Daarom wordt op basis van deze evaluatie één open-source model gekozen om te finetunen. In deze fase zal ook verder worden gebouwd aan de basis UI die is ontwikkeld in 1.2.

*Figuur 1: Voorstel NVIDIA GPU cluster*  


* 1. Fine tunen open source model

In deze laatste fase fine-tunen we het model voor onze doeleinden. We bekijken de vragen die frequent aan de SSC-ICT servicedesk gesteld worden en trainen het model om deze goed te beantwoorden. Dit kan ofwel doordat wij voorbeelden meegeven van goede antwoorden ofwel door gebruik te maken van reinforcement learning with human feedback (RLHF) waarbij we stagiairs uit het CFI de kwaliteit van verschillende antwoorden laten beoordelen en op basis van die feedback het model verder optimaliseren.

1. Azure chabot

Om het voor de organisatie makkelijk en aantrekkelijk te maken om AI tools te adopteren wordt ook gekeken naar mogelijkheden die aansluiten op bestaande projecten en infrastructuur, met name binnen Microsoft Azure. Azure biedt de mogelijkheid modellen van OpenAI (bv. GPT4) aan te passen voor een eigen use case. Het biedt ook de mogelijkheid om deze chatbot on-premise te draaien. Je kan hierbij een knowledge base, system prompt en few-shot examples meegeven.

Om de chatbots vergelijkbaar te houden, gebruiken we dezelfde use-case als voor de open-source-optie. De ‘system prompt’ is een omschrijving van wat het systeem (de chatbot) is en moet doen. Door trial-and-error zullen we zoeken naar een prompt die goed werkt voor onze use-case. Few-shot examples betekent dat je hem een paar voorbeelden geeft van hoe je wilt dat hij een vraag beantwoordt, wat het systeem vervolgens kan extrapoleren naar andere vragen. Hierbij zullen we weer de meest gestelde vragen aan de SSC-ICT servicedesk gebruiken.

Het resultaat is een Chatbot in de Azure omgeving met vergelijkbare functionaliteit als de chatbot die we op basis van open-source architectuur maken. Dit stelt ons in staat beide chatbots te vergelijken en de voor- en nadelen van beide aanpakken te beschrijven.

1. Eindrapportage: vergelijking Azure Chatbot met open-source model

3.1 Beoordelingscriteria model vaststellen en ontwikkelen

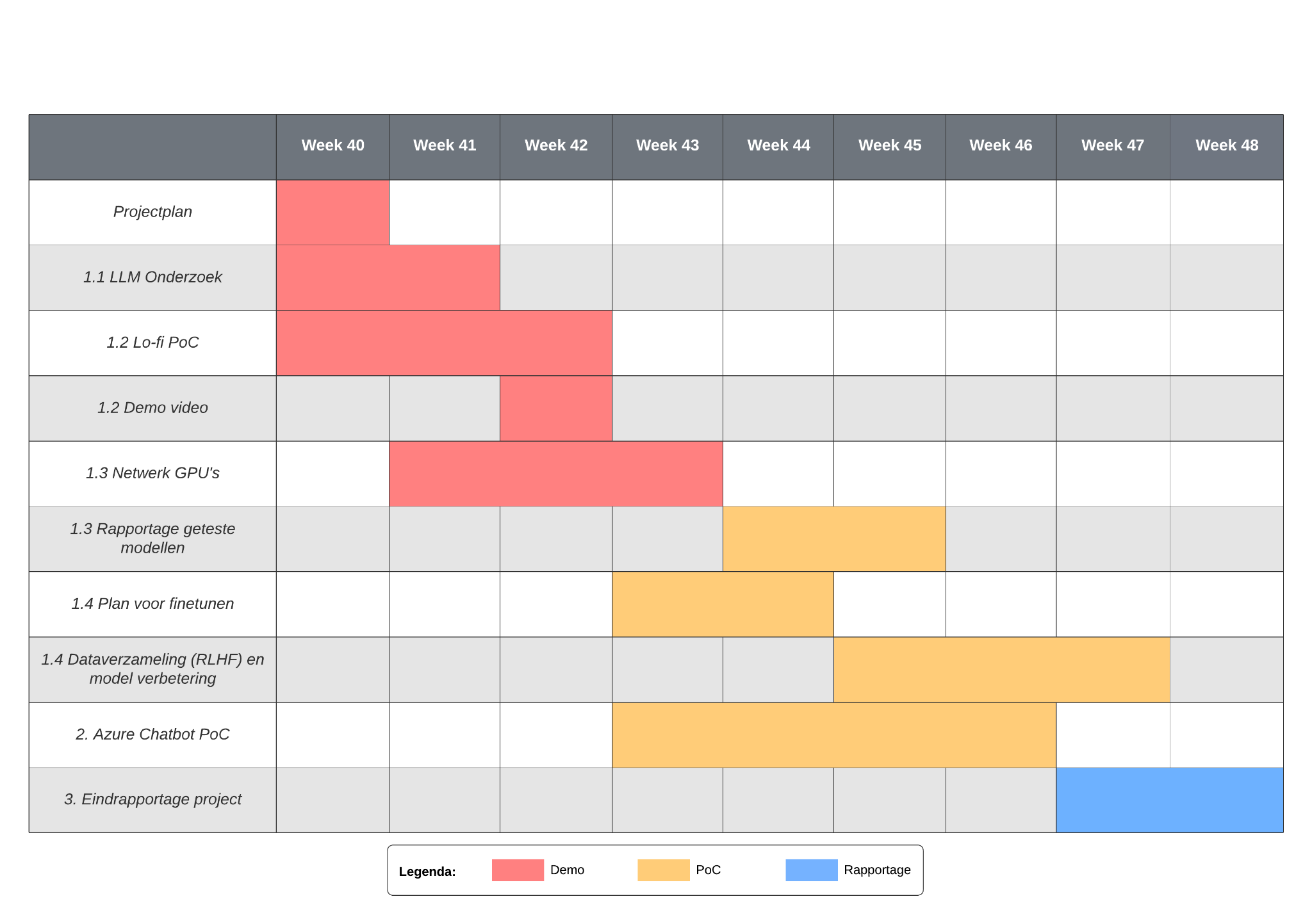
Om een zinvolle vergelijking te kunnen maken tussen een Azure Chatbot en een Open-Source model, is het cruciaal om eerst de criteria vast te stellen waarmee we de prestaties van deze chatbots kunnen beoordelen. Een voorbeeld hiervan zou kunnen zijn hoe effectief ze de 100 meest gestelde vragen aan de SSC-ICT helpdesk kunnen beantwoorden. Daarbij is het van belang dat we begrijpen wat onze stakeholders belangrijk vinden in een chatbot. We zullen proberen om meetbare aspecten te identificeren of te ontwikkelen. Het uiteindelijke resultaat zal worden gepresenteerd in de vorm van een tabel waarin meetbare verschillen tussen de twee chatbots worden weergegeven, zoals performance, kosten of energieverbruik, voor zover we deze gegevens kunnen verkrijgen.

3.2 Overige verschillen en afwegingen

Het vergelijken van een open source benadering met een kant-en-klaar product, zoals een Azure chatbot, gaat verder dan alleen prestaties en energieverbruik. Er zijn fundamentele verschillen tussen deze benaderingen, zoals de mate van afhankelijkheid van externe partijen, flexibiliteit en de mogelijkheid om ze naadloos te integreren in andere diensten die je mogelijk gebruikt. Ons uiteindelijke rapport zal ook aandacht besteden aan deze bredere aspecten van de vergelijking.

Het eindproduct zijn 2 chatbots en een rapport waarin deze twee opties met elkaar vergeleken worden.

Planning & Team

*Figuur 2: Gantt diagram van planning week 40 - 48*  


|  | **Deelonderzoek** | **Deliverables** | **Datum** |
| --- | --- | --- | --- |
| 1.1 | LLM Onderzoek | Rapport over beschikbare open source LLM’s | 6 okt |
| 1.2 | Lo-Fi PoC | Chatbot draaiende op een CPU, met een basis interface | 20 okt |
| Demo video PoC | 27 okt |
| 1.3 | Lokal chatbot assistent | Netwerk van GPU’s | 1 nov |
| Rapport geteste modellen | 10 nov |
| Doorontwikkelde UI | 15 nov |
| 1.4 | Fine tunen model | Plan voor finetunen | 1 nov |
| Data verzameling en model verbetering | 24 nov |
| 2 | Azure Chatbot | PoC | 7 nov |
| 3.1 & 3.2 |  | Eindrapportage project | 8 dec |

*Tabel 1: Deliverables en planning*

| **Naam** | **Functie** | **Rol** | **Periode** |
| --- | --- | --- | --- |
| Jelle van Elburg | I-Trainee | Projectleider | 1 aug - 1 dec 2023 |
| Jitse Goutbeek | I-Trainee | Technisch Projectleider | 1 sep - 30 apr 2024 |
| Mathijs de Niet | Stagiaire MBO Software Developer | Stagiaire | 1 sep - 30 apr 2024 |
| \**De full-time equivalent (FTE) inzet voor dit project komt uit het CFI en betreft daarom tijdelijk inzetbare werknemers* | | | |

*Tabel 2: Samenstelling Team\**

1. Een quantized model is een gecompresseerde, lichte versie van een model met verminderde precisie [↑](#footnote-ref-0)
2. LangChain is een open source framework voor het maken van LLM applicaties [↑](#footnote-ref-1)
3. NVIDIA AGX Xavier Developer Kit (32GB) [↑](#footnote-ref-2)