

Technisch ontwerp

Project : DI.ON.NE

Opdrachtgever : Marcel Schmitz

Auteur : Tygo Offermanns

Datum : 11-09-2024

Versie : 1.1

Inhoud

1.	Inleiding	1
2.	De oplossing in kaart	1
2.1.	Beschrijving van de oplossing.....	1
2.2.	Technische randvoorwaarden.....	2
3.	Overzicht technische opbouw	2
4.	Applicatie configuratie	3
4.1.	Installatie	3
4.2.	Configuratie / systeem instellingen.....	3

1. Inleiding

In deze sectie wordt een diepgaande analyse gegeven van de technische aspecten van het DI.ON.NE-project. Het hoofddoel is om inzicht te bieden in de architectuur, infrastructuur en implementatie van de applicatie. Dit is van cruciaal belang om het prototype succesvol te laten functioneren, onderhouden en verder uit te breiden.

De applicatie bestaat uit een virtuele avatar, ontwikkeld in Unity, die gebruikers kan ondersteunen door middel van tekst- en spraakinteractie. De applicatie draait zowel op desktop als mobiel, en kan ook gebruikt worden in Augmented Reality (AR).

2. De oplossing in kaart

2.1. Beschrijving van de oplossing

De oplossing omvat de ontwikkeling van een interactieve applicatie met een digitale avatar die studenten kan ondersteunen bij vragen en leerdoelen. De avatar reageert op zowel tekstinput (via een invoerveld) als spraakinput (via een microfoon). Deze input wordt verstuurd naar een lokaal draaiend Large Language Model (LLM), waarna de avatar een antwoord genereert en dit weergeeft in de interface.

De architectuur bestaat uit drie lagen:

1. Unity-interface: toont de avatar, biedt de gebruikersinterface (UI) en registreert input.
2. API-laag (Flask): fungeert als schakel tussen Unity en de AI-modellen.
3. AI-modellen (Ollama LLaMA 3 en Whisper): zorgen voor tekstgeneratie en spraakherkenning.

Daarnaast ondersteunt de oplossing AR-projectie van de avatar, zodat deze zichtbaar kan worden gemaakt in de fysieke omgeving van de gebruiker.

2.2. Technische randvoorwaarden

Voor de succesvolle werking van de applicatie zijn de volgende randvoorwaarden vastgesteld:

- **Unity-omgeving:** Unity wordt gebruikt voor de visuele representatie en gebruikersinteractie.
- **Avatarcreatie:** de avatar wordt gemaakt met Ready Player Me en aangepast aan voorkeuren van studenten.
- **Communicatie-API:** een Flask-server verwerkt verzoeken van Unity en communiceert met de LLM.
- **Spraakherkenning:** Whisper van OpenAI wordt geïntegreerd om spraak naar tekst te vertalen.
- **Tekstverwerking:** LLaMA 3 van Ollama genereert antwoorden op basis van tekstinput.
- **Cross-platform ondersteuning:** de applicatie moet draaien op desktop (Windows), mobiel (Android) en AR via Unity AR Foundation.
- **Privacy en lokaal gebruik:** alle data en AI-modellen draaien lokaal, zodat gebruikersdata niet extern wordt verwerkt.

3. Overzicht technische opbouw

De applicatie is opgebouwd uit verschillende componenten:

- **Programmeertaal en engine:** Unity (C#) voor de client en Python (Flask) voor de server.
- **Gebruikersinterface (UI):** invoerveld, microfoonknop en avatarweergave in 3D. De interface is responsive en geschikt voor desktop en mobiel.
- **API:** Flask met CORS-ondersteuning. Endpoints zijn onder meer /ask (Text-to-Text) en /voice (Speech-to-Text).
- **AI-modellen:**
 - **LLAMA 3 (Ollama)** voor Text-to-Text.
 - **Whisper** voor Speech-to-Text.
- **Dataflow:**
 - Gebruiker typt of spreekt input in Unity.
 - Input gaat naar de Flask API.
 - De API stuurt de input door naar het juiste AI-model.
 - Het antwoord gaat via de API terug naar Unity.
 - De avatar toont het antwoord in de interface.
- **AR-functionaliteit:** geïmplementeerd met Unity AR Foundation en een AR-camera.

- **Schaalbaarheid:** de code is modular opgezet zodat functies als Text-to-Speech, emotieherkenning en koppeling met leermanagementsystemen later kunnen worden toegevoegd.

4. Applicatie configuratie

4.1. Installatie

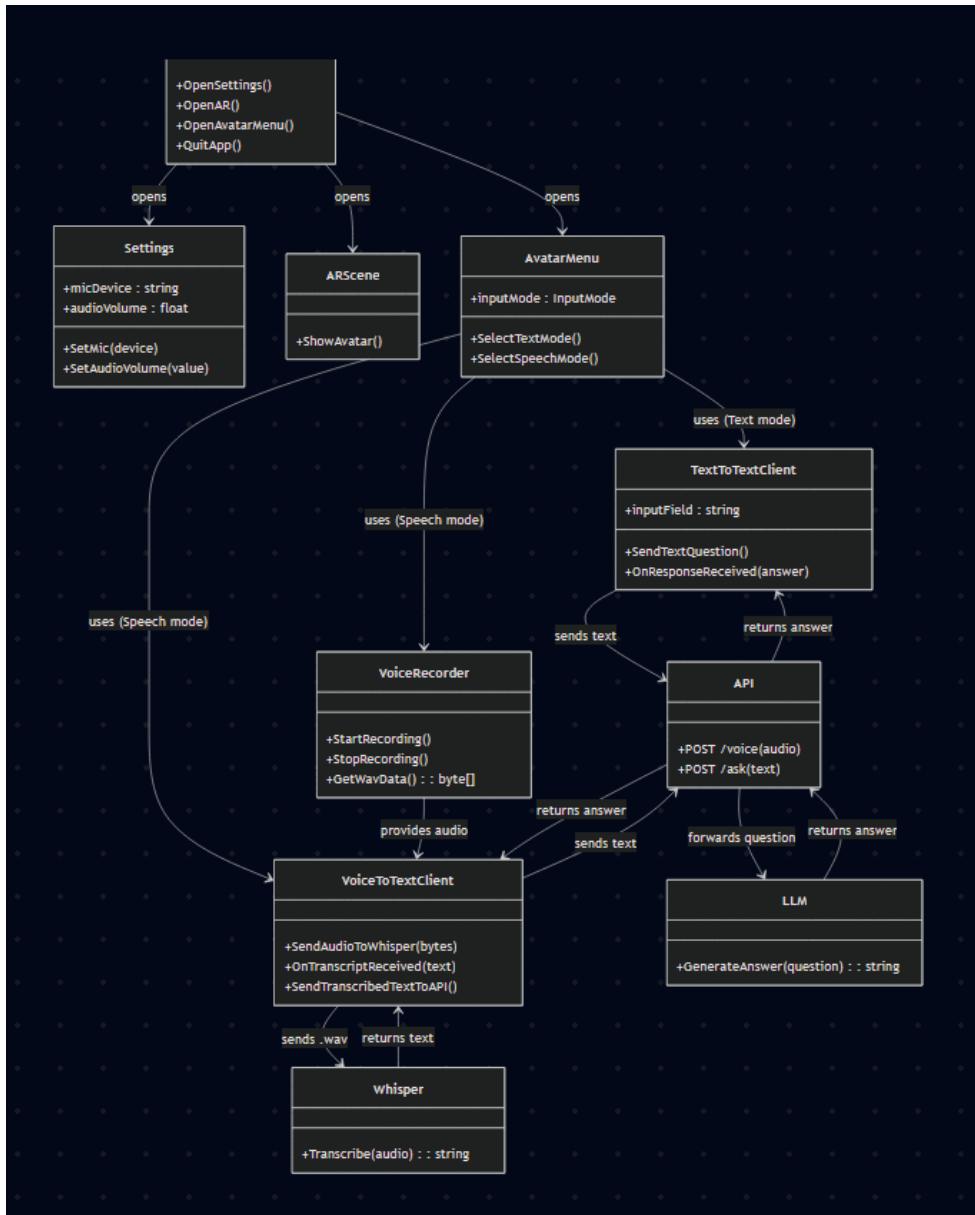
- Installeer Unity (versie 2022 of hoger).
- Download de benodigde Unity-pakketten (AR Foundation, Input System).
- Clone of download de projectbestanden van GitHub.
- Installeer Python (3.10 of hoger).
- Installeer de benodigde Python-bibliotheken:
 - pip install flask flask-cors ollama
 - pip install openai-whisper
- Start de Flask API met:
 - Python api.py
- Open Unity en koppel de juiste scripts (TextToTextClient, VoiceToTextClient, SceneHandler).

4.2. Configuratie / systeem instellingen

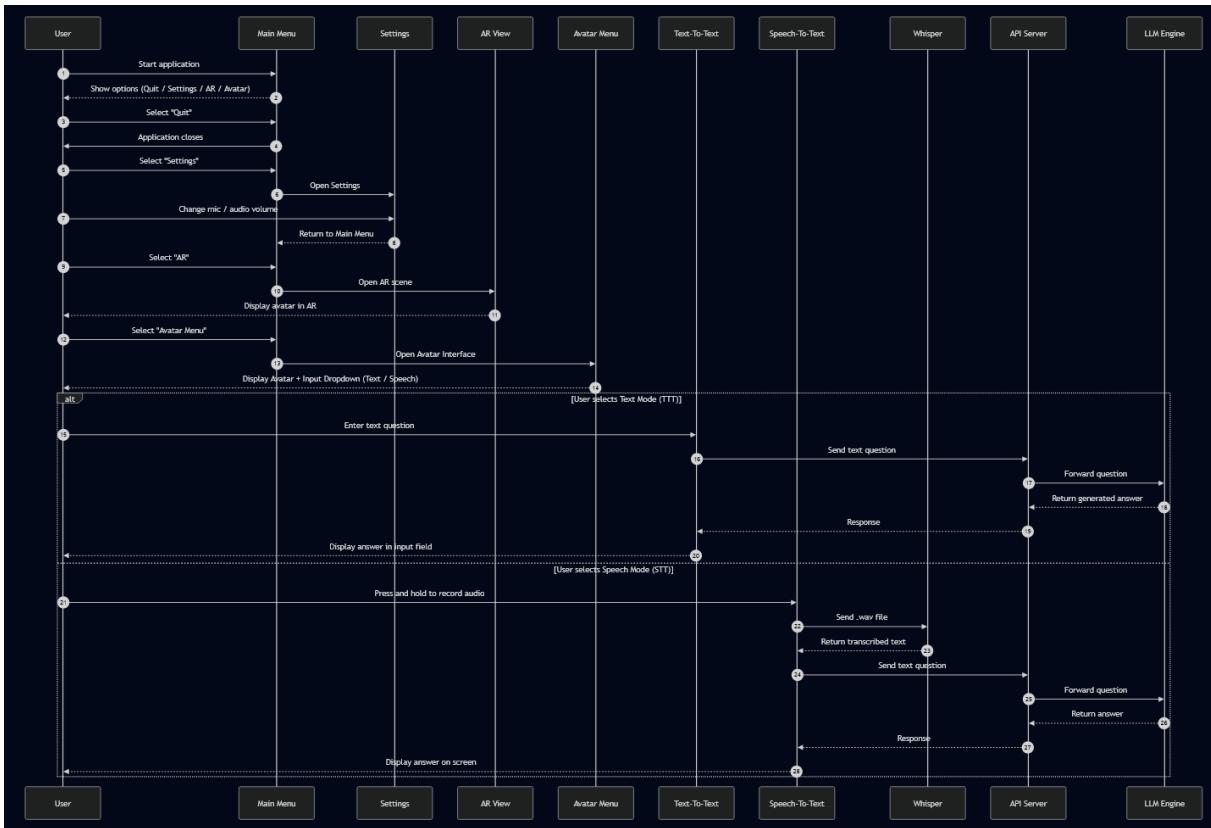
- In Unity moeten build profiles ingesteld worden voor Windows Desktop en Android Mobile.
- Voor AR moet AR Foundation worden ingeschakeld en gekoppeld aan de AR-camera.
- In de API moet de juiste LLM geselecteerd worden (Ollama met LLaMA 3).
- Whisper moet worden geconfigureerd voor spraakopnames in .wav-formaat.
- Het canvas moet ingesteld worden op responsive voor mobiele schermen.

5. Technische tekening

Class diagram:



Activiteiten diagram:



Flowchart:

