Team Members (alphabetized by last name):
Markson Chen (1009002598)
Yibin Cui (0000000000)
Barry Jiang (0000000000)
Kevin Liu (0000000000)

Assignment A1: Project Landscape

Part One: Interest Statements

Our team aims to build an interactive virtual character system driven by multimodal foundation models, focusing on real-time interactability that combines speech, text, and motion. As a team, we are excited to explore how conversational AI, vision understanding, and motion generation can be combined into a cohesive, low-latency system that feels engaging in virtual spaces.

Kevin I have a background in training and fine-tuning LLMs in modalities such as natural language, symbolic music and chemistry. Therefore, I want to fine-tune VLMs for virtual-idol-centered interactions and work on the memory retrieval system, with an end goal of reviving my favourite VTubers Xiangwan and Jiale.

Barry My background is in embodied AI, and I have aspired to build an AI VTuber like Neuro-sama that can play games while streaming. In this project, I hope to work on vision-language navigation and high-level actions.

Yibin I am particularly interested in multimodal AI and human-AI interaction, and through my prior work on chatbots and translation systems, I have developed an understanding of sequence modelling and memorization in neural networks, which I am excited to apply toward integrating language models with speech and real-time interactive pipelines.

Markson As the proposer of this project, I wish to not only recreate and interact with my favourite AI streamer, Neuro-sama, but also use this as an opportunity to expand my tech stack in embodied intelligence, as my field of research primarily involves modelling embodied intelligence in computational neuroscience.

Part Two: Landscape Analysis

Relevant Item	Description	Commentary
Whispers From the Star	Anuttacon is an AI start-up focused on AIGC, especially for games. There first product, Whispers From the Star, is a AI-powered interactive story game powered by LLMs.	It showcases how LLMs can drive dynamic, narrative-rich game experiences, making it relevant as an example of interactive AI. However, it is primarily a scripted story engine rather than a real-time embodied agent, and it does not integrate continuous multimodal perception or embodied motion.
Grok companion & Project AVA	Grok provides voice-to-voice AI companions with 3D avatars. These are the same software behind Razer Project AVA.	Grok provides 3D avatar based AI, which is very similar to what we are building. However, grok companion cannot move around, they just stand there.
Paper: DualVLN	A dual-system VLN foundation model that synergistically integrates Qwen2.5 7B VL for high-level reasoning, generates pixel goals, and converts to low-level action execution via a small trajectory diffusion model.	Traditional SLAM-based navigation produces static maps and is unable to handle dynamic scenes. On the other hand, VLM can understand conversational context as well as navigational spatial intelligence to an extent, making them suitable for our project.
Paper: DartControl	A diffusion-based autoregressive motion generation framework that enables real-time, text-driven control of humanoid motion. DartControl generates continuous full-body motion conditioned on high-level language instructions and low-dimensional goals, while maintaining temporal coherence and low latency suitable for online interaction.	DartControl is directly relevant to our project as it provides a practical solution for real-time, goal-directed motion generation. However, it does not model rich scene interactions or semantic understanding of the environment, requiring upstream perception and planning modules, which is an opportunity our system aims to address.

HY-Motion (Tencent Hunyuan)	A diffusion-based text-to-motion generation framework designed for high-quality and controllable human motion synthesis. HY-Motion focuses on generating diverse, stylistically rich full-body motions from natural language prompts, emphasizing motion realism and expressiveness.	HY-Motion demonstrates strong capabilities in generating expressive, high-quality motion from text, making it relevant for virtual character animation. However, it is primarily designed for offline or batch generation and lacks explicit support for real-time, goal-directed interaction, which limits its direct applicability to interactive embodied agents without additional control and planning layers.
Open-LLM-VTuber	An open-source framework for building AI-powered virtual streamers and VTubers. The project integrates large language models with speech recognition, text-to-speech, voice conversion, and avatar control, providing an end-to-end pipeline for interactive AI characters.	Open-LLM-VTuber is highly relevant as a practical reference for assembling multimodal VTuber systems. However, its architecture is primarily designed for streaming and turn-based interaction, with limited support for low-latency embodied interaction, real-time motion planning, and environmental understanding, highlighting opportunities for more tightly integrated perception–action loops.
AIAvatarKit	An open-source toolkit for building AI-driven 3D avatars. AIAvatarKit provides tools and pipelines for integrating language models, animation systems, and avatar controls to enable conversational and interactive 3D characters in games and virtual environments.	AIAvatarKit serves as a flexible, open source foundation for AI avatar development, making it relevant to projects that combine conversational AI with animated 3D characters. However, it focuses primarily on avatar integration and pipeline tooling rather than real-time multimodal perception, embodied goal-directed behavior, or navigation planning.
Pipecat	An open-source low-latency streaming orchestration framework for multimodal pipelines, optimized for continuous speech → text → LLM → text → speech loops and real-time multimodal interaction.	Pipecat provides a powerful foundation for building real-time conversational pipelines and has been selected as the starting orchestration layer for our project. However, we need additional modules in our architecture for

	Pipecat also integrates with various STT, LLM, TTS, and vision services, and supports transport layers like LiveKit.	continuous perception, world understanding, and goal-directed motion.
Qwen3-Omni	A natively end-to-end multilingual omni-modal foundation model capable of understanding and generating across text, image, audio, and video inputs with real-time streaming responses. It uses a unified architecture to process diverse modalities and can produce both text and natural speech outputs without compromising single-modal performance.	Qwen3-Omni represents the leading direction in unified multimodal AI and is relevant to our project's needs for integrated perception across modalities. However, it is a general foundation model and does not itself provide domain-specific world understanding, semantic mapping, embodied motion control, or navigation planning.
Neuro-sama	Neuro-sama is an AI-powered virtual streamer that interacts with audiences on livestream platforms through real-time conversation and gameplay. It leverages large language models to generate dialogue and respond dynamically to chat, presenting itself as a virtual character with a persistent online presence.	Neuro-sama demonstrates the appeal and viability of AI-driven virtual characters in social and entertainment contexts. However, its interaction is primarily text- and voice-based, with limited embodiment and environmental awareness. The character does not engage in real-time spatial interaction or goal-directed motion, highlighting an opportunity for our project to extend AI companions into fully embodied agents that can perceive and act within shared virtual environments.
Paper: Enhancing Robotic Collaborative Tasks Through Contextual Human Motion Prediction and Intention Inference	A research proposing a deep learning architecture that jointly predicts future 3D human motion and human intention in collaborative human–robot tasks. The model incorporates task-specific contextual information using multi-head attention and is validated on real-world datasets for human–robot handover and collaborative harvesting scenarios.	This work highlights the importance of context and intention awareness in human–robot collaboration, which aligns with our project's emphasis on socially aware interaction. However, it focuses on predicting human behavior for robot planning, rather than enabling an embodied agent to act and interact in real time.

Part Three: Project Outline

Problem statement

A solution for interactive virtual characters in 3D has yet to emerge as of today, and we believe the core engineering challenge is designing a cohesive, low-latency, multimodal architecture that can (1) support modular SOTA conversational models end-to-end, (2) maintain continuous visual grounding, (3) convert binocular visual input into reliable navigation and motion plans, (4) preserve and retrieve memory across sessions, and (5) orchestrate these asynchronous processes into a consistent, engaging, real-time agent experience.

Proposed solution

Our AI system will consist of 4 modules:

1. A speech-to-speech pipeline that includes STT (Speech-To-Text), LLM inferencing, TTS (Text-To-Speech), that supports most of the SOTA local/cloud-based models, and can integrate with the current LLM roleplay ecosystem and voice-changing ecosystem.
2. A vision understanding module that continuously feeds the textual description of the current frame for LLM inferencing. If suitable model exists, it can be replaced by a video understanding module that feeds the textual description of the recent video vision for LLM inferencing.
3. A motor module that integrates a vision-language navigation model, a 3D understanding model, and a goal-directed motion generation model to achieve motor planning and control from only binocular video input
4. A memory module, implemented using existing solutions like Mem0 (or Letta).

Aside from the AI models, our project also includes:

- An orchestrator based on message queues that asynchronously routes all the models into one coherent system.
- An interface layer that sends text, speech and motor output from the model into VRChat, and sends binocular video (with audio) from VRChat to the model. It could also support other SteamVR games or game engines like Unity.
- An application GUI that manages all the settings, parameters, and additional controls

Project Milestones

Week 0 (Jan 11 - Jan 17)

- Draft the system architecture and project plan (Finished)

Week 1 (Jan 18 - Jan 24)

- Implement OpenXR driver interface to send body, head, and hand motion into SteamVR and get binocular video back.
- Implement a demo of message queue-based chatbot backend and (perhaps) integrate with Pipecat. Run the full speech→text→LLM→text→speech pipeline.
- Implement API calls to send input into VRChat, like character emotion, text, and avatar height
- Find and run a small Vision Language Model for vision understanding
- Explore the feasibility of semantic mapping from vision for navigation

Week 2 (Jan 25 - Jan 31)

- Finalize the speech-to-speech backend (if not finished or harder than expected)
- Integrate the VLM as real-time visual descriptions into the pipeline
- Integrate a navigation-focused VLM for generating 2D goal coordinates
    - Generate RGB-D from binocular video, and use it to map 2D goal coordinates into 3D global coordinates (as an input into the motion generation model, DART)
- Try DART for avatar motion generation (while navigation is simultaneously being developed)
    - Implement a module that continuously plans for low-level motion for DART

Week 3 (Feb 1 to Feb 7)

- Integrate our current system with the LLM roleplay ecosystem based on character cards
- Finalize the work on navigation if not finished
- Tune the LLM to output an appropriate emotion label
- Exploration of letting the VLM to control the head (camera) movement and output the gaze location

Week 4 (Feb 8 to Feb 14)

- Implement agent memory using Mem0
- Finalize the entire motion module from low-level control to high level planning
- Explore motion stylization approaches
- Implement voice changing methods (RVC, GPT-SoVITS, and the better ones)

Week 5 (Feb 15 to Feb 21)

- Optimize the whole system for latency. Explore quantization, distillation, or other methods.
- If time allows, explore the possibility of replacing single-frame vision understanding with low-latency video understanding
- Design and code the application GUI

Week 6 (Feb 22 to Feb 28)

- Write an entire wiki site for people to use our project
- Test multiplatform support

Week 7, 8, 9

- Buffer time in case we overestimate the complexity of our project.
- If we have finished everything on time, we can explore additional features.

Unknowns to investigate

1. Among all the possible options, how should we use OpenXR/OpenVR to bidirectionally connect Unity and VRChat with our model inputs and outputs, aiming for ease of development and a low-latency system?
2. Is it more cost-efficient to start from open-source projects (like ALXR and Monado) that might save us time yet introduce undesired complexity for our project, or should we leverage coding agents to build everything ourselves?
3. Is it actually technically feasible to integrate the vision-language navigation model, 3D understanding model, and goal-directed motion generation model into a coherent motion generation system (nobody has ever done it before)?
4. What is the best system architecture for our entire system (not overly-complicated, integratable with existing ecosystems, and has low-latency)?

Part Four: Project Press Release

Introducing OpenNeuro: Interactive Characters in Virtual Worlds - Powered by Artificial Neural Networks

“OpenNeuro brings lifelike AI characters to life, letting users meet, talk with, and interact with vivid virtual characters that move, react, and share space with them in real time.”

For decades, virtual characters have lived behind the screens. They are mesmerizing yet can only stay as memories. OpenNeuro changes that experience. Today, users can put on a VR headset or open a desktop application and meet any fictional character face-to-face. The character listens, reacts, and moves naturally in response to speech, gestures, and what happens in the environment.

OpenNeuro is an interactive AI-driven virtual character system built for real-time multimodal interaction. It combines conversational AI, speech processing, visual perception, and motion generation into a single, responsive pipeline. This allows characters to coordinate voice, gaze, and body movement as users speak and act in a shared 3D space. Users can ask a character to step closer, react to a gesture, or focus on an object nearby and see that intent reflected immediately through lifelike behaviour.

The Customer Problem

As conversational AI has improved, expectations have grown with it. Users now want AI characters to behave like social entities, not detached interfaces. Most existing systems still fall short. They rely on text or voice alone and lack awareness of space, movement, and user actions. Even advanced virtual characters and livestream avatars often depend on scripted animations or delayed responses, which quickly break immersion.

In VR platforms, virtual social spaces, and livestream environments, this gap makes interaction feel stiff and limited. Simple actions such as sharing attention, reacting to gestures, or moving naturally within a scene are often missing or feel awkward and artificial.

The OpenNeuro Solution

OpenNeuro closes this gap by treating language, perception, and action as parts of the same interactive loop. Conversation, vision, and movement are continuously connected, allowing the

system to decide how a character should respond moment by moment. The result is a vivid, embodied AI character that can participate naturally in virtual spaces, responding with expressive speech and motion instead of static or scripted behaviour.

Key Benefits

- **Embodied interaction:** Characters see the environment and respond to speech, gestures, and visual context
- **Expressive presence:** Coordinated voice, gaze, and body movement make characters feel lifelike and alive
- **Flexible deployment:** Works across VR platforms, desktop applications, and livestream settings
- **Fully Open Source:** Built for research, experimentation, and creative development

What The Users Are Saying

"It feels less like using an AI tool and more like being in the same space as a character that actually reacts to me."
— Early VR User

"OpenNeuro is revolutionary technology and years ahead of the field."
— Embodied AI Researcher

"This project shows how AI characters can move beyond scripted behavior and become almost… ALIVE."
— Project Stakeholder

"I was not emotionally prepared for how good this project is."
— Vedal

Who Is It For

OpenNeuro is designed for VR users, VTuber creators and audiences, researchers exploring AI and the edge of technology.

As virtual worlds become more social and immersive, OpenNeuro enables a new generation of AI characters that can see, move, and respond alongside users in real time. This opens new possibilities for creativity, interaction, and experimentation inside shared digital spaces.

Appendix

Iteration 1

We initially described OpenNeuro as an AI virtual idol focused on entertainment and live-streaming.

Target Audience: Otakus

Key Features: An AI virtual idol powered by LLM with a 3D avatar. The idol is accessible as 1. a local running AI companion (mostly for debug) 2. A player in VRChat 3. A channel on a streaming platform like Twitch. The target audience can interact with the model through text, voice and movement through 1. streaming platforms such as Bilibili, Twitch, etc. 2. VRChat 3. A locally running application.

Fictional quotes:

Vedal: “This novel, open-source, end-to-end AI virtual idol is too good! Neuro-sama cannot compete against it😠”

Iteration 2

In this iteration, we sought to improve perceived realism, with a focus on rapid system responsiveness, facial expression control, and natural movement.

Introducing OpenNeuro - A Real-Time Embodied AI Companion for Virtual Worlds

We are excited to introduce OpenNeuro, an embodied AI companion designed to interact with users naturally in virtual environments. Unlike traditional chatbots or static AI avatars, OpenNeuro combines conversation, perception, and motion into a real-time system that allows it to exist and respond within shared virtual spaces.

OpenNeuro is built for platforms such as VRChat, desktop applications, and livestream environments, where interaction goes beyond text and voice. By responding to speech, gestures, and visual context with coordinated voice, gaze, and body motion, ### creates the experience of a virtual character that feels socially present.

The Customer Problem

As conversational AI improves, users increasingly expect AI characters to behave like social entities rather than disembodied interfaces. However, most AI companions today lack awareness

of physical space and user actions. Even AI-powered virtual idols are often limited to scripted animations and delayed responses, breaking immersion in interactive environments.

For users in VR and virtual social platforms, this disconnect makes natural interaction difficult. Simple actions such as asking AI to move closer, react to a gesture, or share attention toward an object are either unsupported or feel unnatural.

Our solution

OpenNeuro addresses this gap by integrating language understanding, visual perception, and motion generation into a unified, low-latency pipeline. Instead of treating conversation and movement separately, OpenNeuro continuously reasons over what users say, what it sees, and how it should act.

Users can speak to OpenNeuro, gesture in the environment, or give spatial instructions, and see those intentions translated into expressive behavior. OpenNeuro understands visual context, plans goal-directed movement, and responds with coordinated speech and motion, allowing it to participate naturally in shared virtual space.

Target Audience

OpenNeuro is designed for VR users, VTuber creators and fans, researchers exploring embodied and multimodal AI, and developers interested in open, interactive AI systems.

Key Benefits

- Embodied interaction: Responds to visual context
- Real-time responsiveness: Act with low-latency
- Expressive presence: Emotion and body motion reinforce social realism
- Flexible deployment: Works as a VRChat avatar, desktop companion, or livestream co-host
- Open and extensible: Designed for research, experimentation, and creative use

What Users Are Saying

"It feels less like talking to a chatbot and more like interacting with a character that's actually there with me."
— Early VR User

"OpenNeuro *shows how language, perception, and action can come together in an embodied AI system."*
— Embodied AI Researcher

"OpenNeuro *demonstrates how conversational AI can move beyond text and voice to become something users can interact with in shared space. That shift opens up entirely new possibilities for virtual platforms."*
— Project Stakeholder

"This novel, open-source, end-to-end AI virtual idol is too good! Neuro-sama cannot compete against it"
— Vedal

Iteration 3 (In Part 3)

We realized the core issue is that current AI companions cannot naturally interact with users in shared virtual space. This final version focuses on how our product makes interaction feel more natural and present.