PersonaKit (PK): A Plug-and-Play Platform for User Testing Diverse Roles in Simulated Full-Duplex Dialogue

Hyunbae Jeon
Department of Computer Science
Emory University
Atlanta, GA, USA
harry.jeon@emory.edu

Jinho D. Choi
Department of Computer Science
Emory University
Atlanta, GA, USA
jinho.choi@emory.edu

Abstract

As spoken dialogue systems take on diverse personas—authoritative instructors, uncooperative merchants, distracted workers—they need distinct, human-like turn-taking to stay immersive. Yet current full-duplex systems default to a rigid "always-yield" policy during overlap, undermining character consistency for non-submissive roles, and evaluating persona-specific alternatives through user studies demands substantial real-time engineering. We present PersonaKit (PK), an open-source, low-latency web platform that *simulates* full-duplex turn-taking pragmatics—via a cascade pipeline and prompt-level control rather than a jointly trained speak-while-listening model. Through intuitive JSON configurations, researchers define personas, specify probabilistic interruption handling (yield, hold, bridge, override), and auto-deploy comparative A/B surveys. An in-the-wild evaluation with 8 personas shows that PK is an extensible, end-to-end framework for studying complex sociolinguistic behaviors in next-generation spoken agents.

1 Introduction

The shift of spoken dialogue systems from half-duplex to full-duplex (Skantze, 2021; Ma et al., 2025) moves the frontier of immersive-agent design from text quality to *sociolinguistic behavior*: how agents manage overlapping speech. In natural conversation, turn-taking is heavily mediated by interpersonal roles and status (Sacks et al., 1974). An authoritative military instructor handles an interruption very differently than a subservient virtual assistant. Yet most commercial full-duplex systems default to an *always-yield* policy, immediately ceding the floor on user speech and breaking immersion for non-submissive roles. Testing how conversational pragmatics affect user perception requires significant engineering effort across WebRTC audio, Voice Activity Detection (VAD), dynamic LLM prompt injection, and latency tracking. We developed **PersonaKit (PK)** to eliminate this bottleneck. Importantly, PK is full-duplex at the audio-transport layer—it captures speech continuously, even while the agent speaks—but deliberately *simulates* full-duplex turn-taking pragmatics via a cascade and prompt-level control rather than a jointly trained speak-while-listening model, trading architectural fidelity for zero-engineering configurability. PK contributes (i) an open-source web platform for simulated full-duplex persona evaluation, (ii) a JSON-based mechanism making persona-conditioned interruption policies first-class configurable objects, and (iii) an end-to-end workflow from live deployment through auto-generated surveys to structured log export.

2 Related Work

Full-duplex spoken language models (Ma et al., 2025) have magnified the need for robust turn-taking modeling (Skantze, 2021), but classic systematics (Sacks et al., 1974) remain hard to integrate into LLM-driven agents. Persona modeling (Zhang et al., 2018), meanwhile, is predominantly evaluated in text, ignoring the acoustic pragmatics of dominance, yielding, and barge-in recovery. PK bridges these threads by exposing turn-taking strategy itself as a persona parameter and evaluating it in live voice interaction.

3 System Architecture

PersonaKit isolates low-latency audio engineering from experimental design: researchers configure everything through a web dashboard and four JSON files—for the persona (role and opening prompt), the interruption strategy matrix, the survey, and LLM/TTS routing. The stack is open

Code: github.com/emorynlp/PersonaStudyKit
Demo: persona-studykit.run.app
Video: youtu.be/EBpF_TSNXbw

Proceedings of the 27th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 597–601
August 2–5, 2026.

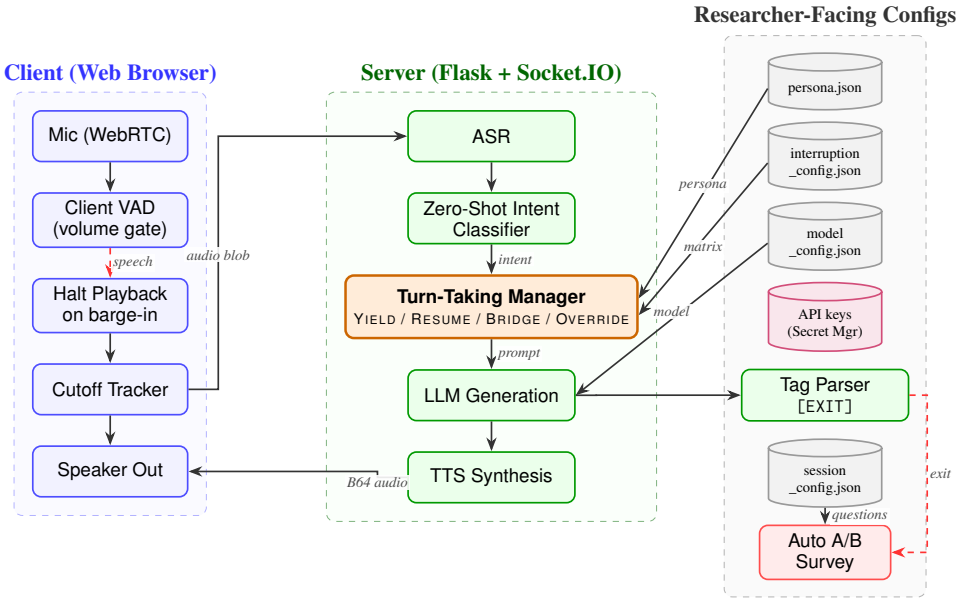


Figure 1: PersonaKit architecture. The browser performs client-side VAD and logs the *cutoff text* on barge-in; the server classifies interruption intent, and the Turn-Taking Manager selects a strategy (Yield / Resume / Bridge / Override) from `persona.json` and `interruption_config.json`. All experimental behavior is configured through JSON—no code is modified.

source (Python/Flask + vanilla JS), so PK can be cloned, run locally, or pointed at new providers.

A simulated, not native, full-duplex stack. PK records continuously and halts playback the instant the user speaks, so the channel is genuinely bidirectional. The agent, however, does not *process* speech while talking: on barge-in it runs a cascade (halt playback → ASR → intent classification → strategy selection → generation). PK thus *simulates* full-duplex turn-taking pragmatics with cascade components and prompt engineering, not a jointly trained speak-while-listening architecture—a deliberate choice that lets researchers manipulate turn-taking entirely through configuration, at the cost of native concurrency.

Client-side VAD and audio tracking. The frontend uses WebRTC for microphone capture with a client-side VAD node that halts local playback on barge-in. PK tracks byte-level playback to log the *cutoff text* (what the bot vocalized before being interrupted) and the *remaining text* it still intended to say; both return to the server so the LLM knows precisely where it was interrupted.

Turn-taking as a persona tool. When interrupted, the server transcribes the user's utterance and classifies its intent into four categories grounded in prior phonetic and conversation-analytic work (Cao et al., 2025): **Competitive** (seeks the floor to contradict or override), **Cooperative** (adds information without derailing), **Topic Change** (pivots subject), and **Backchannel** (short affirmations, not floor-taking). Researchers define a strategy matrix in `interruption_config.json` mapping these intents to four actions (**Yield**, **Resume**/Hold, **Bridge**, **Override**) with probability weights—e.g., a dominant persona might weight Competitive interruptions as 50% Resume, 25% Override, 15% Bridge, 10% Yield, while a cooperative persona inverts these. These four actions mirror strategies documented in human interruption handling—ignoring the interruption and continuing, acknowledging it while signaling intent to retain the floor, or yielding—which prior work shows are deployed differently depending on a speaker's role and the interrupter's intent (Cao et al., 2025).

How probabilities drive generation. The matrix is applied *before* generation: on each interruption the Manager reads the intent, samples an action from its categorical distribution, and injects it as a control token into the system prompt (e.g., "`[STRATEGY=RESUME]`: finish your previous sentence, ignoring the user"), so the LLM generates *conditioned on* a pre-committed action. The Autonomous condition (Style C) skips sampling and lets the LLM choose zero-shot.

Automated lifecycle and data export. Sessions end on a `MAX_TURNS` cap or a verbal `TERMINATE` in-

tent; the LLM emits an in-character farewell with a hidden `[EXIT]` tag that triggers the post-session survey. Each session exports the dialogue transcript, an event log with per-turn intent, strategy, and cutoff/remaining text, and all survey responses as JSON or CSV.

Quadrant	Personas
Q1: High Agency, Low Comm.	Drill Sergeant, Tavern Keeper
Q2: High Agency, High Comm.	Salesperson, Tour Guide
Q3: Low Agency, High Comm.	AI Assistant, Librarian
Q4: Low Agency, Low Comm.	DMV Clerk, Distracted Chef

Table 1: Persona catalog mapped to the Interpersonal Circumplex (Wiggins, 1979).

4 Pilot User Study

Study Design. Five participants (N=5) completed the full study, engaging with 8 occupational personas balanced across the Interpersonal Circumplex (Table 1), yielding 120 dialogue sessions. The catalog deliberately spans the full circumplex—from high-agency, low-communion roles (e.g., Drill Sergeant) to mild, cooperative ones (e.g., Tour Guide, Librarian)—and our goal is to demonstrate that persona-conditioned turn-taking can be *configured and tested* within PK, not to validate any particular persona-to-strategy mapping. For each persona, users experienced three randomized within-subject conditions: **Style A** (Always-Yield baseline), **Style B** (Probabilistic, JSON-tuned strategy weights), and **Style C** (Autonomous, where the LLM selects a strategy zero-shot from the persona prompt). Condition order was randomized per persona and the underlying LLM and voice were held fixed across styles.

Evaluation Metrics. After each persona, participants completed a comparative Likert survey on $\{-1, 0, +1\}$ rating Reaction Naturalness (*"felt human-like and natural"*; Bartneck et al., 2009), Persona Consistency (*"remained consistent with the role"*; Gomes et al., 2013), and Interaction Fluidity (*"turn transitions felt smooth"*; Skantze, 2021), followed by a forced-choice preference item and a free-text justification. Because each item elicits a *relative comparison* between conditions rather than an absolute attitude, we use a coarse 3-point comparative scale $\{-1, 0, +1\}$; comparative judgments of this kind are more reliable and impose lower cognitive load than absolute ratings (Thurstone, 1927), at the cost of fine-grained resolution. Default end-to-end barge-in latency under our OpenAI/ElevenLabs configuration was $\sim$1–2 s.

	Style	Nat.	Cons.	Flu.	Pref. (%)
Q1	A (Yield)	0.20	0.70	0.50	20
	B (Prob.)	**0.60**	0.60	**0.70**	20
	C (Auto.)	0.30	**0.70**	0.60	**60**
Q2	A (Yield)	0.50	**1.00**	**0.70**	40
	B (Prob.)	**0.60**	**1.00**	**0.70**	**50**
	C (Auto.)	0.30	0.70	0.60	10
Q3	A (Yield)	**0.50**	**1.00**	**0.90**	**70**
	B (Prob.)	0.30	**1.00**	0.70	10
	C (Auto.)	0.30	0.90	0.70	20
Q4	A (Yield)	0.50	0.80	0.60	**50**
	B (Prob.)	**0.67**	**0.90**	**0.70**	30
	C (Auto.)	0.30	0.80	0.60	20

Table 2: Mean ratings on $\{-1, 0, +1\}$ for Naturalness, Consistency, and Fluidity, plus forced-choice preference (Pref. %), by circumplex quadrant and style (N=5; 10 ratings per cell). **Bold** = highest mean per quadrant, not a significant difference.

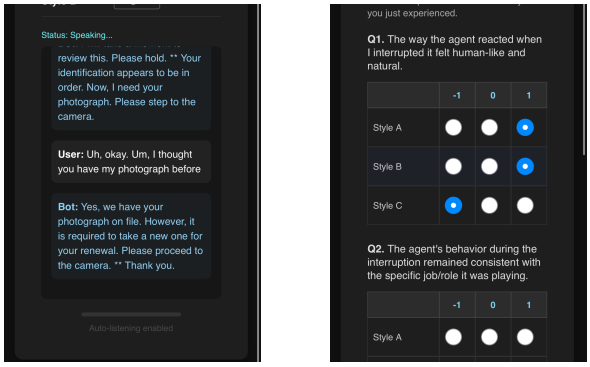
Results. Table 2 reports per-quadrant means across five completers (10 ratings per cell). Two patterns emerge. **High-agency personas (Q1)** appear to benefit from non-yielding strategies: Reaction Naturalness rises from 0.20 (Yield) to 0.60 (Probabilistic), and 60% of forced-choice votes favored Autonomous. **Low-agency, high-communion personas (Q3)** tended to favor yielding, with 70% preferring Always-Yield. Q2 preferred Probabilistic (50%), and Q4 preferred Yield (50%) yet reached its highest naturalness (0.67) under Probabilistic. These are descriptive tendencies rather than significant effects; per-persona logs are released with the repository for finer-grained analysis.

Emergent Interruption Behaviors. PK's survey engine auto-collects free-text justifications that surface persona-consistent reasoning: for the Drill Sergeant, P1 liked that the agent "*ignored me when I interrupted…just like a real boot camp*"; for the Tour Guide, P3 preferred Style B because "*it was trying to finish what it was saying*"; and for the Tavern Keeper, P4 "*had fun with C*" as the agent wavered between meal and ale. The raw logs reveal behaviors Always-Yield would erase: in one Probabilistic Drill Sergeant session, the line "…*Repeat it again!*" was cut at "*Repeat it*"; classified `COMPETITIVE` and sampled `RESUME`, it resumed "…*again!*"—a coherent barge-in recovery.

5 Demonstration and Use Cases

At SIGDIAL, attendees experience PK from both sides: they re-route turn-taking by uploading a new JSON, then speak through a laptop or phone (Fig-

ure 2) and try to interrupt a *Grumpy Tavern Keeper* (holds the floor) versus a *Standard AI Assistant* (yields)—feeling how interruption policy reshapes perceived role realism even when the LLM is unchanged.



(a) Live dialogue view (b) Auto-deployed survey

Figure 2: PersonaKit runs on desktop and mobile. **(a)** Participant view: turns from each side, persona/style labels, live VAD status. **(b)** Post-session comparative survey, auto-generated from `session_config.json`.

Beyond the study, PK is a general-purpose testbed for full-duplex dialogue research. **Persona prototyping:** iterate on the persona prompt, turn-taking matrix, and scenario in `persona.json` and run a live study immediately—the primary workflow. **Custom surveys:** `session_config.json` accepts arbitrary Likert, forced-choice, and free-text banks for other constructs (e.g., trust, task success). **Model comparison:** routing in `model_config.json` lets LLM vendors, voices, or local open-weight models be swapped while persona and policy stay fixed. **Data collection:** the event log pairs each interruption with its intent, sampled strategy, and follow-up utterance—a seed set for supervised barge-in policies or RLHF reward models.

6 Limitations and Conclusion

Our pilot (N=5) is descriptive, not inferential: with 10 ratings per cell, the bolded entries in Table 2 mark the highest observed mean, not a significant difference, and larger, cross-demographic samples are needed before stronger circumplex-to-strategy claims. The scale is also coarse—several Consistency cells hit the $+1$ ceiling, so a finer scale would be needed to resolve top-of-range differences in a larger study. The persona-to-strategy matrices are *designer-specified hypotheses*, not mappings validated against real interaction data; validating them against human–human and human–agent corpora is a priority for the planned full-length study. Intent classification uses an unvalidated zero-shot prompt and can mislabel ambiguous back-channels under noise. Finally, the four-action vocabulary excludes fine-grained prosodic cues such as pitch reset, latching, and gaze—a deliberate tradeoff favoring configurability over acoustic fidelity. Despite these limits, PersonaKit exposes turn-taking as a JSON-configurable persona parameter and automates the full study lifecycle from recruitment to export. Our pilot (N=5) suggests that preferred turn-taking policies may vary with persona role, illustrating PK's usefulness as a testbed; PK is open source and ready for community extension.

References

Christoph Bartneck, Dana Kulić, Elizabeth Croft, and Susana Zoghbi. 2009. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International Journal of Social Robotics*, 1(1):71–81.

Shiye Cao, Jiwon Moon, Amama Mahmood, Victor Nikhil Antony, Ziang Xiao, Anqi Liu, and Chien-Ming Huang. 2025. Interruption handling for conversational robots. In *Proceedings of Robotics: Science and Systems (RSS)*.

Pedro Gomes, Carlos Martinho, and Ana Paiva. 2013. Metrics for character believability in interactive narrative. In *Interactive Storytelling*, pages 92–103. Springer.

Ziyang Ma, Yakun Song, Chenpeng Du, Jian Cong, Zhuo Chen, Yuping Wang, Yuxuan Wang, and Xie Chen. 2025. Language model can listen while speaking. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Harvey Sacks, Emanuel A. Schegloff, and Gail Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4):696–735.

Gabriel Skantze. 2021. Turn-taking in conversational systems and human-robot interaction: A review. *Computer Speech & Language*, 67:101178.

L. L. Thurstone. 1927. A law of comparative judgment. *Psychological Review*, 34(4):273–286.

Jerry S. Wiggins. 1979. A psychological taxonomy of trait-descriptive terms: The interpersonal domain. *Journal of Personality and Social Psychology*, 37(3):395–412.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of ACL*.

Equipment Requirements for Demonstration

PersonaKit runs in any modern browser on a laptop or phone, so requirements for the demo are minimal.

Equipment provided by authors. A laptop with built-in microphone and speakers, running PK via its cloud-deployed instance; a phone as a secondary client to show mobile compatibility; and a pair of over-ear headphones that attendees can use if the demo area is noisy.

Furniture and equipment requested from organizers. A standard demo table (one attendee at a time, plus a presenter), two power outlets, and reliable conference Wi-Fi. No projector or external monitor is needed.

Interaction flow. The demo is open to all attendees: each walks up, speaks with the agent through the provided laptop or phone, and (on request) switches between a *Grumpy Tavern Keeper* and a *Standard AI Assistant* to experience how persona-conditioned turn-taking changes the feel of the interaction. A consent notice is shown before each session, and any transcripts retained briefly for on-site walkthroughs are deleted at the end of the demo.