Build-A-Being Yourself: A web-based system for authoring embodied conversational agents in healthcare research

Christopher You[1][0000−0002−2045−5796], Rashi Ghosh[1][0000−0002−7297−9911], Andrew Maxim[1][0000−0001−9346−4717], Xuening Peng[1][0000−0001−8749−3212], Danish Tamboli[1][0009−0006−5850−1133], Zhanhong Huang[1][0009−0000−7238−6780], Ryan Canales[1][0000−0003−0554−7817], Roshan Venkatakrishnan[2][0000−0002−6538−627X], and Benjamin Lok[1][0000−0002−1190−3729]

[1] University of Florida, Gainesville FL, USA {christopheryou, rashighosh, amaxim, xuening.peng, danishtamboli, zhanhonghuang, r.canales}@ufl.edu, lok@cise.ufl.edu

[2] University of Central Florida, Orlando FL, USA roshan.venkatakrishnan@ucf.edu

Abstract. Embodied conversational agents (ECAs) can extend the reach and impact of healthcare professionals when time and attention are limited. However, these benefits are best realized when domain experts are actively involved in the design, authoring, and deployment of ECAs. This paper contributes the Build-a-Being Yourself (BABY) system, a platform designed to help domain experts directly create entire ECA conversations through a no-code interface. We adopted an end-user development approach to create a low-threshold, high-ceiling tool for building ECA conversations, informed by collaborations with domain experts. BABY enables rich design and customization through integrated tools that allow end users to design ECA models, create multi-modal rule-based and LLM-supported conversations, and automate deployment to web-enabled devices. We describe our process of end-user design, the BABY system, and a brief validation on use and scalability with domain experts and two published user studies. Alongside, we present findings on our process and describe opportunities for innovation for end-user tools for broader HCI research.

Keywords: Embodied conversational agents · System · End-user development · healthcare.

1 Introduction

A substantial body of human–computer interaction (HCI) research has explored how user health and education can be supported by *embodied conversational agents* (ECAs), which are computer-generated characters that interact with users through verbal and nonverbal behaviors [37,27,30]. Advances in graphics, artificial intelligence (AI), and large-language models (LLMs) [30] have increasingly enabled ECAs to serve as conversational partners for training healthcare professionals [11,32,31,50] and to empower patients to take charge of their health [35,47,13,55]. The benefit of ECAs lie in their ability to extend human reach and impact while reducing barriers of scale, access, and cost. However, ECA designs must be informed by healthcare domain experts if their benefits are to be fully realized [36]. Although numerous tools have been developed to streamline the process of building ECAs [23,2,20], many are designed primarily for programmers or users with substantial technical expertise. Among those intended for less technical domain experts [40,48,21,56], existing tools can still be difficult to employ, offer limited support for web-deployment and interaction, or lack designs grounded in the needs and workflows of the target users.

Therefore, this paper contributes to ECA healthcare research by introducing a system that empowers healthcare domain experts to design and deploy ECAs. The development of this system is guided by an *end-user development* (EUD) approach, which HCI research identifies as a strategy for supporting tools that balance ease of use with expressive power for domain experts acting as end users [33]. Our system is grounded in two EUD concepts: (1) *low-threshold* [42], lowering technical barriers in employing ECA conversations, and (2) *high-ceiling* [42], robustly supporting diverse use cases and sophistication in their design. As a result, we present the ***Build-A-Being Yourself*** (BABY) system: a no-code, feature-rich tool that enables domain experts to design multimodal ECA interactions compatible with LLMs. These ECAs can then be deployed on web-enabled devices, allowing healthcare domain experts to both build ECAs and conduct research grounded in their own expertise.

We present the BABY system by first describing our EUD-informed participatory design approach, in which healthcare domain experts' existing workflows and prior ECA experiences informed the system's development. We then detail how this feedback was translated into concrete system features and design decisions. A brief validation is provided with five domain experts through think-aloud sessions paired with semi-structured interviews. In doing so, we also demonstrate BABY's scalability and research applicability through its real-world deployment and use in publications in two HCI venues. Ultimately, this work contributes a system that extends ECA development to domain experts, facilitating innovative healthcare-informed ECA designs, rapid prototyping, and scalability in research.

2 Related Work

This section reviews prior work informing BABY's design across two dimensions. First, we examine existing ECA systems and tools, comparing their technical foundations and conversation design paradigms to BABY. Second, we review end-user development (EUD) approaches, particularly template and wizard-based strategies, that enable non-technical users to create software and shape BABY's design philosophy. Across both areas, we highlight how BABY builds

	Designing ECAs Does the tool enable option *presets* or *customized* granularity?		Authoring Conversations Does the tool enable *rule*, *retrieval*, or *generative* approaches, with *LLMs*?		Infrastructure Does the tool enable game *engine independence* and *web-deployment*?	
Tool Name	**ECA Models**	**Convers-ational Behavior**	**Design Approach**	**LLMs**	**Engine Indepen-dent**	**Web-Deployed**
VHTK [23]	Presets, Custom	Presets, Custom	Rule, Retrieval, Generative	✓	×	×
AUP [2]	Presets	Custom	Rule, Retrieval	×	×	✓
VAIF [20]	Custom	Custom	Rule	×	×	×
VPF [48]	Presets	Presets	Retrieval	×	×	✓
VIP [56]	Presets	Presets	Rule	×	×	✓
HealthDial [5]	Presets	Presets	Rule	×	✓	×
BABY	**Presets, Custom**	**Presets, Custom**	**Rule, Generative**	✓	✓	✓

Table 1: Generalized comparison of existing ECA authoring tools and BABY. Tools are classified according to how *end users* must utilize this system to interact with the six described feature spaces (e.g., end users can theoretically design custom ECAs to upload to VIP, but by default it relies on existing modules).

on, diverges from, and complements prior work to support domain experts in creating ECA interactions without extensive programming expertise.

2.1 Prior ECA Systems

This section reviews key ECA systems and tools that have laid the groundwork for ECA development. While generalized comparisons are made across systems to BABY (see Table 1), we note that many systems have evolved since their inception and instead focus on core design characteristics rather than exhaustive features. The reviewed systems are organized into two groups: those primarily designed for technical researchers and developers, and those intended to support domain experts with limited technical expertise. Systems are discussed regarding *designing ECAs*, *authoring conversations*, and underlying *infrastructure*.

Technical End Users. End-user tools are often designed for more technically-capable personnel rather than lay persons. Beyond easing the burden for such users, these approaches raise the ceiling for interaction, enabling the creation of more flexible and expressive systems through standardized frameworks (e.g., SAIBA, FML, BML, GRETA) and robust NLP/NLU components. For instance, the **Virtual Human Toolkit** (VHTK) is a pioneering system whose initial design employed a modular structure for ECA design, authoring, and infrastructure [23,19]. ECAs and their behaviors are generated using preset models and customizable components like *SmartBody*, *NonVerbal Behavior Generator*, and markup languages. Drawing on Hussain et al.'s classifications for conversational agent designs (i.e., rule-based, retrieval-based, or generative-based) [26], VHTK conversations are largely rule-based and retrieval-based using natural language understanding and generation with integration to interpret non-verbal information. Similarly, **Agents United Platform** (AUP) is built on several modules to perform a series of *sense*, *remember*, *think*, and *act* steps to generate configurable conversational behaviors and conversations [2]. Dialogues use a WOOL Web Service, enabling rule-based and retrieval-based conversations, with GRETA and ASAP supporting preset ECA options. The **Virtual Agent Interaction Framework** (VAIF) enables users to create ECA conversations and scripted events within a single Unity application, often for virtual and augmented reality [20]. VAIF provides developers with baseline tools and configurations for creating rule-based conversations; in turn, developers must define their own ECA models and specific conversational behaviors.

These systems demonstrate robust capabilities for creating ECA conversations, but remain challenging for non-technical end users to adopt. BABY aims to lower the threshold for designing ECA conversations via a web-based interface to author conversations directly without prior setup or dependencies. Conversations are built through an interface wizard that provides preset configurations for ECA properties, such as models, voices, poses, moods, and behaviors. It is worth noting that both VHTK and AUP have introduced alternative versions of their systems, recognizing the challenge for less technical personnel [21,22]. However, the described systems require computationally demanding game engines like Unity; BABY instead reduces technical overhead by rendering ECAs exclusively through JavaScript libraries, thereby also improving device compatibility. For a visual comparison to VHTK, AUP, and VAIF, see Table 1.

Domain Expert End Users. Prior research has also explored tools that enable domain experts to directly participate in designing ECA conversations. For instance, University of Florida researchers created the **Virtual People Factory** (VPF) to create virtual patients through crowdsourced dialogues from healthcare experts and novice users [48,34,53]. In VPF, a suggestion system prompts experts to provide potential user inputs, creating a corpus of responses for retrieval-based ECA conversations. The same group also created the **Virtual Interviewer Platform** (VIP) to create rule-based educational conversations, especially to promote healthy behaviors through persuasion models [56,55]. Both platforms utilize largely preset ECA models and conversational behaviors (e.g., voice, non-verbal behaviors), with extensions or modifications typically requiring developer intervention; however, they also allow domain experts to directly create and/or deploy ECA conversations on web-enabled devices. These two systems reflect recent developments in ECA authoring; Nouraei et al.'s **HealthDial**

enables healthcare providers to directly author educational dialogue (similar to VPF) for rule-based conversations (similar to VIP) [40]. This work builds on Bickmore et al.'s relational agents, which were originally designed for more technically oriented researchers [6,4,5,3]. HealthDial provides preset models and conversational behaviors which may be extended using tools from relational agents, markup languages, and animation toolkits. Generative approaches help create dialogues, but the system is intentionally designed to support structured, rule-based interactions to shield from LLM-generated errors.

These systems demonstrate how domain experts can directly create or influence the design of ECA conversations, especially in healthcare research. BABY aims to raise the ceiling of these systems by enabling domain expert end users to customize ECAs directly through a web interface using integrated libraries and APIs (e.g., Avaturn, MetaPerson, Mixamo, ReadyPlayerMe); largely, any Mixamo-rigged model can be used in BABY (see Section 3.2). Furthermore, BABY was designed to empower domain experts to utilize LLMs in their practices for generative conversations. Noting the concerns about LLMs in HealthDial, we enable domain experts to use LLMs alongside controlled, rule-based educational conversations that can be augmented with retrieval; we also allow domain experts to pilot-test purely prompt-driven virtual patients. Finally, conversations created in BABY are automatically deployed to the web via load-balanced instances and are accessible from web-enabled devices, with the goal of maximizing the reach and impact of ECA interventions (see Table 1).

2.2 End-User Development

EUD is an HCI approach to design tools that allow non-technical users to create, modify, or extend software [33]. The intention is to utilize EUD to help our domain experts, or end users, develop tools that are *low-threshold* with a *high-ceiling* [39]. User interface tools should allow novices to learn how to use them and build confidence as they do. Simultaneously, as users advance in skill, they should be able to develop increasingly sophisticated applications. EUD encompasses various approaches designed to empower non-expert users to create software applications independently [33]. Two prominent EUD strategies for this article are template-based development [1] and wizard-based development [9].

Template-based methodology is an approach in EUD where end users fill in, select, or modify predefined templates to create or customize software. For instance, AgentSheets uses rule-based templates to allow end-users to design agent behaviors as part of block-based programming [45]. Similarly, He et al. proposed a template-based approach to end-user model-based software engineering, enabling users to create valid models without having to define complex constraints [24]. This methodology not only simplifies the development process but also enhances flexibility, allowing users to effortlessly customize features and content to their needs while avoiding underlying details [1].

Wizard-based development, another key strategy in EUD, guides users through the software creation process step by step using intuitive wizards. By breaking complex tasks into manageable subtasks, this approach ensures users remain focused on the current step rather than being overwhelmed by the entire workflow. Wizard-based development proves particularly effective when integrated with other methodologies, as demonstrated by Castelli et al., who employed it to customize data visualizations in smart home environments [10]. Additionally, Fogli et al. used wizard-based development to enable novices to develop e-government services [17]. Through its structured guidance and simplified task breakdown, wizard-based development empowers users to create functional applications without requiring extensive technical knowledge.

BABY combines template-based and wizard-based development strategies to empower non-technical, domain experts in creating ECA conversations. By leveraging template-based flexibility and intuitive wizard-based guidance, BABY supports the customizable, scalable design of multi-modal ECAs. This dual-strategy approach lowers entry barriers and facilitates rapid prototyping, allowing domain to bring their creative visions to life in the virtual world.

3 System

3.1 End-User Development Design Approach

BABY's design is motivated by internal collaborations with healthcare domain experts with whom we have partnered in developing ECAs to help patients navigate cancer screening [57,29,52], learn about clinical trials [29,8], assess nutrition and diet [51], and address mental health goals [54]. Accordingly, we incorporate feedback from collaborators with strong content expertise but no experience in animation pipelines, game engines, or ECA architectures. We derived patterns from collaborators' designs of ECA conversations and pilot interviews of their workflows to construct a low-threshold, high-ceiling tool. These recurring design patterns were abstracted into concrete design constructs within BABY [42] to transform observed ECA authoring practices into reusable system elements aligned with end users' reasoning. Abstract design challenges were addressed by integrating primarily *wizard-based* and *template-based* compositions in BABY [15,33]. We identified three system modules where these compositions could be concretely realized: (1) designing ECA models and behaviors, (2) authoring ECA conversations, and (3) supporting deployment through research-ready infrastructure. Each section describes BABY's design for a low-threshold, high-ceiling tool derived from observed practices and constraints in collaborators' ECA workflows, and provides brief implementation details that operationalize these abstractions.

3.2 Designing ECAs

EUD Design in BABY. We identified the need for flexible and accessible ECA design to support the diverse roles ECAs play across research contexts. These use cases require ECAs whose appearances, voices, languages, and expressive behaviors can vary while maintaining appropriate professionalism and plausibility. At the same time, end users reported limited experience with ECA modeling, animation pipelines, or game engines, and noted that existing tools that had previously streamlined their ECA creation (e.g., commercial tools, VPF) often imposed a low ceiling by restricting customization. To address these constraints, BABY adopts a low-threshold, high-ceiling approach to ECA design, using a *wizard-based* interface to guide users through ECA creation.

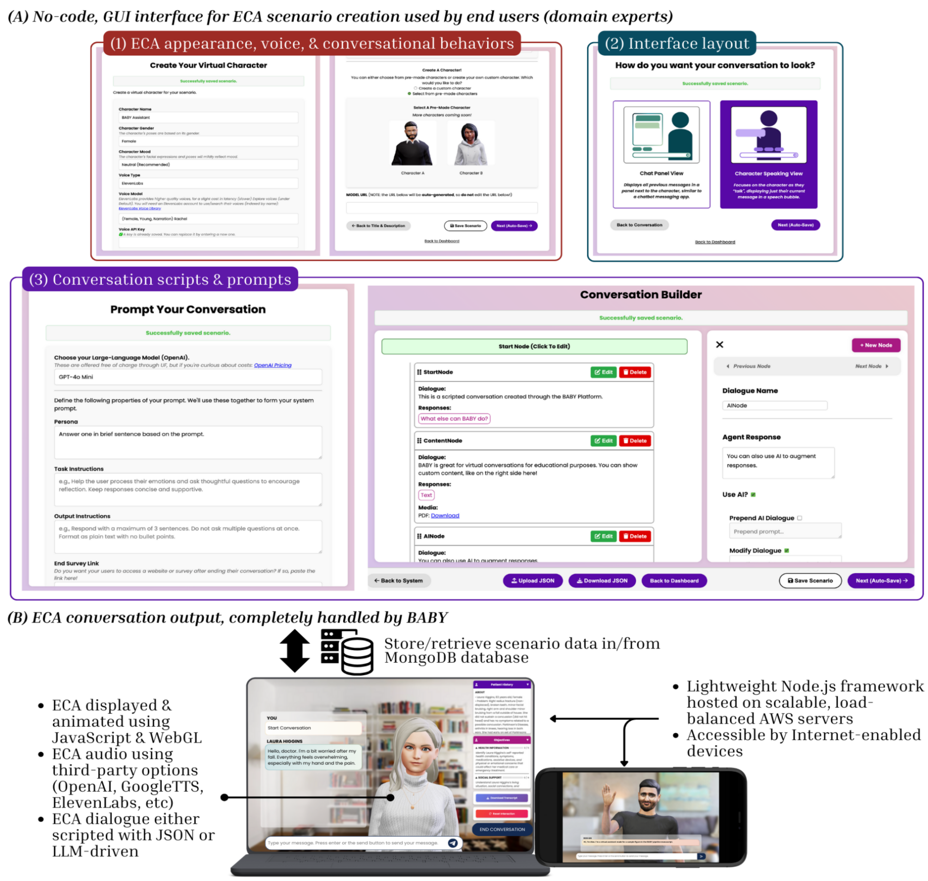

Fig. 1: BABY system overview demonstrating: (A) the design for a low-threshold interface to create high-ceiling ECA interactions by sequentially designing ECAs, templates, and conversations; and (B) the resulting ECA conversations and technical complexity handled by BABY.

BABY enables end users to configure ECA properties to specify moods, poses, and conversational behaviors throughout the conversation. Notably, users can select among preset ECAs or design custom ECA models via BABY's direct integration with third-party avatar creation platforms (see Figure 1 (A-1)). These integrations allow end users to address diverse use cases by specifying more granular customizations, such as hair, facial structure, body shape, skin tone, and clothing. End users can also select preset options for pose and speaking animations (corresponding to masculine or feminine ECA models), the ECA's camera view, and the ECA's mood (e.g., neutral, happy, sad, angry), which influences facial expressions during interaction. For more granular control, the conversation script builder lets users designate animations to ECAs at specific points in the conversation. Voice selection is decoupled from ECA models; this means that users can assign a wide variety of voices from standard, commercial platforms. These also enable end users to create custom voices or select voices tuned for different languages, tones, and expressive qualities.

Implementation. The wizard-based design is built on a web architecture that integrates third-party components for ECA and voice design, as well as conversational behaviors (poses, animations, lip-syncing), abstracting the low-level implementation from end users. To achieve this, BABY builds on the open-source repository that renders and animates ECAs client-side. The repository, *TalkingHead* (**GitHub**: *met4citizen*)[3], leverages a lightweight library called Three.js that performs all ECA functionality exclusively through JavaScript, enabling compatibility with standard desktop and mobile devices. ECAs are represented using compact `.glb` models that bundle geometry and textures into a single file and support blendshapes required for facial animation and lip-syncing. As long as ECAs are Mixamo-compatible and have facial blendshapes (e.g., Avaturn, MetaPerson, ReadyPlayerMe), they can be animated and lip-synced without any supervision from the end user; these tools are also intended to be end-user friendly, requiring no background in 3D modeling to use. Three.js renders the generated ECA models, and *TalkingHead* provides a set of animator functions for lip-syncing, posing, and animating the ECA.

Lip-syncing is achieved by aligning synthesized speech with word-level timestamps to drive viseme-based facial animation, enabling real-time synchronization between audio and ECA articulation. The lip-synced audio is generated in real-time based on the voice assigned by the end user. This removes dependency on a specific voice model, as word-level timestamp transcriptions are used to lip-sync the character. BABY natively integrates models from platforms like OpenAI, ElevenLabs, LemonFox, and Google Cloud, and can be extended for further integrations. *Templates* for non-verbal behaviors, including poses, gestures, idle animations, and moods, are applied using hard-coded animation primitives. However, since the ECAs are Mixamo-compatible, users can simply upload a Mixamo file to their conversation script to trigger an animation at specific conversation nodes. Collectively, these implementation choices support robust ECA creation and integration while preserving extensibility for researchers who wish to incorporate more advanced customization or alternative generation pipelines. Furthermore, these design choices facilitate practical deployment, as ECA model files remain compact (< 5 MB), and computationally intensive audio or text generation is performed server-side while rendering is executed on the client.

[3] https://github.com/met4citizen/TalkingHead

3.3 Authoring Conversations

EUD Design in BABY. End-user workflows revealed a need for the ECA system to enable reliable and bounded conversational behavior. In health contexts, one must be able to curate content directly, constrain conversational scope, and maintain clear control over what information is presented. End users' prior conversation scripts often achieved this through branched logic and rule-based scripts with fully pre-scripted responses and inputs [55,51]. At the same time, collaborators expressed strong interest in extending these workflows with LLMs to support more adaptive interactions. Their ECAs supported multiple roles in health, including patient-facing roles (e.g., health educators, wellbeing counselors) and clinician-facing roles (e.g., virtual patients for interpersonal communication training). To address this, we created a no-code graphical user interface (GUI) that enables end users to directly author conversations while selectively incorporating LLMs in a scoped, controllable way [15].

BABY's no-code GUI allows end users to author conversations by selecting between two conversation styles: rule-based and generative-based [26]. This choice aligns the intended role of the ECA with the desired level of control over conversational behavior. For rule-based conversations, the GUI lets end users author conversations as directed graphs with *nodes* (see Figure 1 (A-3)). Here, a node refers to any unit of conversation containing ECA dialogue, response options, non-verbal behaviors, and/or additional content. Similar to VIP and HealthDial [56,40], end users can configure the ECA's dialogue by writing the verbatim text and/or including prompt-based modifications to prepend, append, or refine the output using LLMs. These responses are generated by the system in real time during conversation, using text and audio for ECA lip-syncing and animation. In addition, end users can specify how their users should respond to the node: open text, buttons mapped to other nodes, multiple-select inputs, and speech. The GUI enables end users to curate content and embed supplementary media, such as images, PDFs, or links, alongside the node's response.

In generative-based conversations, end users can use the GUI to create fully LLM-driven interactions by selecting a target model, defining a persona (tone, knowledge focus, and behavior), specifying task instructions to guide the agent's goals, and applying output constraints (formatting, length). To mitigate LLM-based errors and hallucinations, end users may also upload files that act as a knowledge base for retrieval-augmented generation (RAG). The current design for generative-based conversations is primarily intended for interviews with simulated patients rather than for informational content. Therefore, the GUI enables end users to define medical histories for virtual patients, which directly influence the ECA's responses and are displayed to eventual healthcare-trainee users.

Implementation. BABY abstracts the conversation-creation process by generating an underlying JSON conversation script from the GUI. The GUI is implemented as a node-based authoring interface in which each conversation turn is represented as a discrete node in a directed graph, stored alongside the ECA and any attached resources on BABY's servers. *Start* and *End* nodes explicitly define the entry and exit points of a conversation, while all intermediate nodes reference a subsequent node and satisfy required fields (e.g., agent response, input options). The GUI is built with HTML/CSS/JQuery, enabling extensive DOM manipulation and event handling for user interactions such as clicks, drags, and node reordering. It also performs automatic validation to prevent dead ends and flag unintended infinite loops.

For both conversation styles, BABY stores the scenario details into a MongoDB, and the created nodes are stored as a JSON Script uploaded to an AWS S3 Bucket created for the end user. When ECA interactions are loaded, the scenario details are fetched from MongoDB and used to locate the conversation files in S3 that will be rendered to the user. The created JSON script serves as a finite-state machine that specifies the initial dialogue (as designated by the end user) to be delivered by the ECA. Subsequently, each target user response is directed by the JSON script to the next corresponding node, where dialogue and audio are generated through the aforementioned API calls, supplementary content is displayed, and response inputs for the target user are rendered. For generative-based conversations, BABY also optionally enables modern tool-calling mechanisms afforded by LLMs to monitor user inputs and assess whether certain content has been elicited from the ECA. At runtime, tool calls (e.g., OpenAI Responses API) process user inputs and BABY logs objective-relevant indicators, enabling downstream analysis of conversational outcomes. In sum, the authoring structure allows conversations to be visually designed, while affording end users fine-grained control over dialogue structure, content, and ECA behavior.

3.4 Infrastructure

EUD Design in BABY. End users emphasized that infrastructure choices strongly shape both the accessibility and scalability of ECA deployments. Many existing ECA systems rely on game engines to author or deliver interactions, which collaborators noted introduced substantial barriers due to their lack of experience with such tools and the additional computational and deployment overhead they impose on their target users [56,22,48,20]. Furthermore, collaborators' prior work involved deploying ECAs to diverse populations, including rural and geographically distributed groups. This motivated the need to design sophisticated interactions that are accessible across a wide range of internet-enabled devices for research. To address these requirements, BABY adopts an infrastructure design that avoids reliance on game engines while supporting scalable, web-based deployment and flexible integration with existing research workflows.

BABY enables end users to create ECA conversations on the website without any added overhead through its fully web-based interface. The *wizard-based* and *template-based* tools described in Sections 3.2 and 3.3 are integrated onto a web platform, accessible without any required dependencies. In addition to the capabilities mentioned earlier, end users can select from template interface layouts for the conversation and define static content to display within them (see Figure 1 (A-2) and (B)). When all scenario configurations have been set and published, BABY automatically employs dedicated logging and session databases that record timestamped user and ECA interactions while maintaining session state across interactions, enabling longitudinal studies and recovery from interruptions. BABY also allows end users to attach a survey link at the conclusion of the interaction (e.g., Qualtrics) to support integration with research workflows and post-interaction data collection. Furthermore, all scenarios and configurations created by an end user are visible on a private dashboard, accessible only to the creator and specified collaborators. This enables end users to rapidly update, duplicate, or delete scenarios accordingly.

Implementation. BABY's infrastructure is implemented using a client–server architecture built using Node.js and Express, enabling a clear separation between client-side rendering and server-side processing. ECA rendering, animation, and lip-syncing are handled client-side using JavaScript-based components, while the server manages routing, conversation logic, and secure data handling. The use of JavaScript libraries described in Section 3.2 eliminates reliance on game engines and makes a JavaScript-enabled environment the sole technical dependency. User interactions are transmitted to the server for logging and dialogue generation, after which responses and associated audio are streamed back to the client for real-time rendering (see Implementation from Section 3.3). To support research use, BABY employs dedicated logging and session databases that record user and ECA interactions with timestamps while maintaining session state across interactions, enabling longitudinal studies and recovery from interruptions. These are largely handled using tools like SQL and MongoDB, with authenticated access to their APIs. With a created scenario, BABY automatically handles the technical complexity required to deploy a scalable, web-based ECA conversation (see Figure 1 (B)). Scenario configurations and runtime variables are stored in a database-backed backend, and ECA conversations are deployed automatically as Node.js web applications on a cloud-based, load-balanced infrastructure.

The authors acknowledge that the underlying BABY system may be valuable to other researchers who wish to use or extend its framework. As a result, we've adapted a more developer-centric, "do-it-yourself" approach. A scenario configuration file is provided to manually specify conversation settings, and predefined directories indicate where to upload required assets, such as ECA models and associated media. While authored scenarios must be uploaded and hosted individually, this approach allows researchers to build upon BABY's core capabilities and extend the scope of their conversations. The link to the BABY platform, which includes a request form for access to its tools, is at the end of this article.

4 Domain Expert Evaluation

This section outlines our process for gathering end-user feedback on the design of BABY, including: (1) our methods for brief think-alouds and semi-structured interviews with domain experts, and (2) the resulting findings.

4.1 Method

We evaluated BABY's design with five domain experts serving as end users via a think-aloud protocol paired with a brief semi-structured interview. Each end user had previously been involved in ECA content design, but had no technical background in conversational agent architectures. End users' backgrounds included health communication skill training with virtual patients; mental health, clinical trial, nutrition, and cancer education; and pharmaceutical-related domains.

Rather than evaluating system task performance in isolation, we focus on broader usability through think-alouds and ecological validity across real-world deployments of BABY. To complete the think-aloud, each end user was tasked with creating an ECA conversation based on their existing use cases. The steps to complete were as follows: (1) creating a scenario, (2) creating an ECA, (3) assigning verbal and conversational behaviors, (4) authoring a conversation, and (5) testing the deployed created ECA interaction. End-user participants were encouraged to complete the authoring tasks independently, with minimal instruction, to assess BABY's low-threshold usability. When questions arose, the experimenter provided guidance or hints, which helped identify points of confusion and areas where additional system clarification or support may be needed. Following the think-aloud task, a semi-structured interview was conducted to gather reflections on ease of learning, experiences using the system and resulting ECA conversation, and perceived limitations of the system.

4.2 Results

Each end user successfully created a web-deployed ECA conversation, emphasizing how BABY's web interface enabled them to rapidly prototype and test ECA conversations with various character models, voices, and content. The created scenarios included virtual patients, virtual health educators, and health assistants. We describe the findings of this brief evaluation with respect to our EUD aims of a low-threshold, high-ceiling tool. Table 2 summarizes these findings.

Our findings indicate that BABY largely succeeded as a *low-threshold* tool for end users to author and deploy ECAs. End users described the process as "easy to get started", "easy to understand", and "quick to put together a simple interaction" via the abstracted features detailed in Table 2. The rapid time-to-first-interaction was repeatedly identified as a core strength of the platform. All participants were able to configure ECA appearances and voices, select a conversation structure, and author conversational behavior through the GUI. End users reported a few difficulties in defining and designing an ECA's appearance and voice, and compared the process to survey builders within their existing workflows. One notable observation was that end users reported the wizard-based interface *demystified* ECA functionality and clarified the authoring process to them. Additional opportunities to innovate BABY were identified, particularly around clarifying how ECA components interact and further reducing the learning curve of the GUI script editor (see Section 5.2).

"So much of healthcare is just conversing – this does a great job of it. The fact that there is [audio] as well, and [the process] was fairly seamless."
– EU02, Interpersonal Health Communication Instructor

Our findings also indicate that BABY promoted a *high-ceiling* ECA design and authoring for research applications. When designing ECAs, end users high-

EUD Principle	Abstracted Feature in BABY	Gained Insights
Low-Threshold	(1) **Wizard-based interface** lowers interface complexity and load for non-technical users (2) **GUI conversation builder** provides structured, guided path for authoring conversations (3) **Template-based configurations** supports understandable approach to designing ECAs	(1) May require establishing proper mental models for how ECA components relate to each other (2) Evidence that verbal or walkthrough instruction is needed to support fully independent use
High-Ceiling	(1) **Integrated ECA design** supports customizable ECA configurations, voices, and behaviors (2) **GUI conversation builder** yields opportunities for sophisticated, but controllable conversations, with LLMs (3) **Web-deployed ECA conversations** enables diverse research directions with minimal overhead	(1) Variety of customization options were valuable, but also introduced uncertainty for end users (2) Complex conversation authoring requires feedback mechanisms to assess quality and preview progress

Table 2: Table summarizing the qualitative evaluation of BABY's low-threshold, high-ceiling approach based on the features and gained insights the domain expert think-alouds.

lighted the features in Table 2 to hypothesize new directions for their ECA designs. For example, BABY's process evolved on prior end-user workflows by enabling granular customization of ECA appearances and voices, without requiring human actors for non-verbal behavior and audio. Regarding conversation authoring, one end user expressed skepticism toward AI-driven therapeutic interactions, but reported increased confidence when conversational scope was controlled through rule-based scripting and RAG in BABY. The autonomy afforded by the GUI script editor instilled confidence as conversations were iteratively refined, with some end users comparing the experience to "training" the ECA. End-user insights also suggested opportunities to maximize BABY's capabilities through guided configuration choices and feedback mechanisms for previewing and evaluating complex, LLM-driven conversations.

"I imagine if I have a version like my [diet and screening ECA], I could update statistics, components of the conversation, adjust the [ECA], ... give people exposure to different resources. It would be great for community-based work."
– EU05, Health and Wellness Specialist

In addition to the brief think-alouds, we sought to provide ecological validity on BABY's scalability and utility in research. We deployed ECAs in two separate research studies involving a total of 190 participants, comprising cancer patients and university students. Across both studies, the session management,

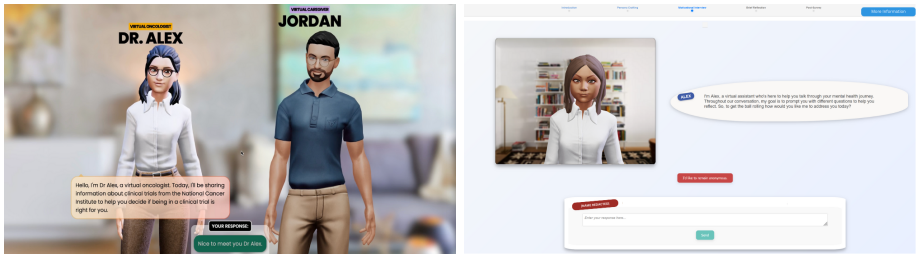

Fig. 2: Published user studies created through BABY: Ghosh et al. integrated multiple ECAs in conversations with cancer patients [18] (left), and You et al. deployed a mental health assistant for college-attending students [54] (right)

logging, and infrastructure components described in Section 3.4 were utilized to deliver the application and collect data. The findings of these studies have been published in peer-reviewed HCI venues (see Figure 2 [54,18]. These deployments provide evidence of BABY's use for study-scale data collection, session management, and cross-device access in real health research. They also demonstrate potential opportunities to expand on the ECA capabilities (e.g., multi-agent conversations, as in [2]) through dynamic interface layouts (see Figure 2).

5 Discussion

In light of BABY and its formative validation, we discuss the role of EUD in designing ECA authoring tools and highlight opportunities for HCI research.

5.1 End-user development

Our process demonstrates that end-user development is a promising approach for designing tools to craft ECA conversations. Abstracting domain experts' workflows into reusable GUI constructs seemingly enabled BABY to maintain low-threshold usability while supporting high-ceiling ECA design. Our work suggests that EUD approaches of *wizard-based* and *template-based* compositions can help lay users design conversational agents and AI tools [15] This aligns with findings suggesting that EUD can empower non-programmers to design data-exploring chatbots [44], social robotic agents [12], and intelligent personal assistants [49]. One notable phenomenon across end users was that building ECAs through BABY helped elucidate the design processes of ECAs and LLMs in shaping interactions. End users indicated that this experience increased their confidence in authoring ECAs independently and improved their understanding of key design decisions for communicating requirements with more technically experienced collaborators. This may suggest that supportive ECA tools can function as both authoring environments and sensemaking aids, echoing prior literature indicating that EUD should promote gradual learning [46] and shared understanding across stakeholders [16].

5.2 Opportunities for HCI Research

Our process also identified opportunities for future HCI research in the design of end-user interactive systems. Across collaborations and end-user evaluations, a recurring theme was a strong interest in integrating LLMs to support dynamic, robust ECA conversations. While end users had prior exposure to LLM tools such as ChatGPT, this familiarity did not necessarily translate into confidence when authoring prompts for conversations. End users frequently expressed uncertainty around what constituted effective conversational design for ECAs. Practical prompting strategies may be helpful for such users [41,38]; however, prior work has shown that end users often develop misaligned mental models of how prompts influence system behavior, limiting their ability to accurately predict outcomes [58]. Similarly, our users indicated difficulty in determining whether a conversation script or prompt had been authored “correctly” or was likely to elicit the intended interactional effects. This suggests a need for HCI research to design end-user authoring tools that go beyond prompt instruction by supporting mental model formation, feedback, and validation of LLM-driven conversational behavior. Zamfirescu-Pereira et al. suggest that abstracting conversational design away from end users may be one approach to mitigate these challenges (e.g., through multi-modal prompting [43]); however, they also identify complementary solutions that help users reason about cause-effect relationships in LLM behavior [58]. Furthermore, even when conversations are reasonably designed, LLM-driven interactions introduced ambiguity in assessing correctness, quality, and alignment with design goals. Prior research has investigated several mechanisms for validating LLMs in conversational AI across domains such as conversational quality, trustworthiness, and accuracy [28,25,14]. However, these evaluation approaches are largely designed for system-level assessment and are difficult to integrate into end-user authoring workflows or to provide actionable feedback during conversation design. This highlights a broader challenge for end-user–oriented AI systems to support users in developing accurate mental models of LLM influence on system behavior and providing mechanisms to validate, preview, or reason about the effects of their design choices.

5.3 Limitations

There are limitations to the system’s evaluation that may better contextualize its contributions. First, although a brief think-aloud study was conducted to help validate the system’s usability, this evaluation was formative and involved a small number of domain experts, which may not capture challenges that emerge with prolonged use or among novice end users. This level of empirical validation aligns with, and in some cases exceeds, prior ECA system work that presents design and implementation contributions without user-based evaluation [2,20,56]. In such cases, the primary contribution is the system itself and the design space it enables. Similarly, our primary contribution is the BABY system, developed through a participatory, end-user development–informed approach that foregrounds domain expert involvement. Second, we did not conduct quantitative comparisons or benchmark evaluations against prior ECA authoring systems. Given the heterogeneity of existing tools and the domain-specific nature of ECA design practices, establishing meaningful, task-equivalent comparisons was not feasible within the scope of this work. Accordingly, we focus on formative, qualitative validation of BABY's design goals and capabilities. The two user studies also aim to provide evidence of BABY's applicability to HCI research. Ultimately, this work demonstrates how an end-user development approach can be operationalized through BABY to support accessible yet expressive ECA design for both applied and research contexts.

6 Conclusion

This work contributes to HCI by presenting a tool that meaningfully bridges innovative ECA research and healthcare research by enabling domain experts to directly participate in building ECA conversations. First, this work contributes a tool that lowers technological barriers to support the rapid development of web-based ECA conversations. Second, we present evidence of BABY's usability through brief think-aloud studies and demonstrate its utility as a research tool in reference to its use in two published empirical user studies. Finally, BABY contributes to HCI research through its EUD approach, which has enabled its use among a variety of educational groups, most notably in university educational curricula [7]. Finally, BABY contributes to HCI research through its EUD approach, which has enabled its adoption across a range of educational contexts, most notably within university curricula [7].

Future work will continue to evolve the EUD process introduced in this study by addressing limitations identified during evaluation, with the goal of further improving BABY's accessibility and expressive capabilities. We also aim to explore new opportunities for designing ECAs by integrating popular retrieval-based conversational approaches designed for end users (e.g., Google Dialogflow), which offer greater control and transparency in conversational behavior. In doing so, we aim to explore how end users can assess and validate their design decisions in inherently subjective processes, such as authoring ECA conversations and LLMs. Finally, we aim to explore how BABY can support collaborative authoring workflows between domain experts and technical collaborators, including versioning, sharing, and iterative refinement of ECA conversations. To this end, we plan to release the underlying system components and reference implementations that support BABY's authoring and execution logic, enabling developers and researchers to adapt and extend the system from a code-centric perspective. Both these materials and the full BABY platform will be accessible by request through the web-based authoring environment to support HCI and ECA research: https://buildabeing.com.

References

1. Barricelli, B.R., Cassano, F., Fogli, D., Piccinno, A.: End-user development, end-user programming and end-user software engineering: A systematic mapping study.

Journal of Systems and Software **149**, 101–137 (Mar 2019). https://doi.org/10.1016/j.jss.2018.11.041

2. Beinema, T., Davison, D., Reidsma, D., Banos, O., Bruijnes, M., Donval, B., Valero, Á.F., Heylen, D., Hofs, D., Huizing, G., et al.: Agents united: An open platform for multi-agent conversational systems. In: Proceedings of the 21st ACM international conference on intelligent virtual agents. pp. 17–24 (2021)
3. Bickmore, T., Gruber, A.: Relational agents in clinical psychiatry. Harvard review of psychiatry **18**(2), 119–130 (2010)
4. Bickmore, T., Schulman, D., Shaw, G.: Dtask and litebody: Open source, standards-based tools for building web-deployed embodied conversational agents. In: Intelligent Virtual Agents: 9th International Conference, IVA 2009 Amsterdam, The Netherlands, September 14-16, 2009 Proceedings 9. pp. 425–431. Springer (2009)
5. Bickmore, T.W.: Relational agents: Effecting change through human-computer relationships. Ph.D. thesis, Massachusetts Institute of Technology (2003)
6. Bickmore, T.W., Schulman, D., Sidner, C.L.: A reusable framework for health counseling dialogue systems based on a behavioral medicine ontology. Journal of biomedical informatics **44**(2), 183–197 (2011)
7. Blue, A., Huynh, L., You, C., Mazorra, G., Black, E., Lok, B.: A patient chatbot to support interprofessional learning (May 2025), https://cab.nexusipe.org/sessions/patient-chatbot-support-interprofessional-learning, conference Presentation
8. Bryan, E.G., Chen, H., Vilaro, M., Chu, H., Grillo, G., Te, P., Buhr, M., Anton, S., Krieger, J.L.: Developing a supportive virtual human to deliver clinical trial education for older women and other populations historically excluded from research. Patient Education and Counseling **130**, 108485 (2025)
9. Caruccio, L., Deufemia, V., Polese, G.: A Wizard Based EUDWeb Development Process, pp. 173–185. Springer International Publishing (Oct 2015). https://doi.org/10.1007/978-3-319-23784-8_14
10. Castelli, N., Ogonowski, C., Jakobi, T., Stein, M., Stevens, G., Wulf, V.: What happened in my home?: An end-user development approach for smart home data visualization. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 853–866. CHI '17, ACM (May 2017). https://doi.org/10.1145/3025453.3025485
11. Cendan, J., Lok, B.: The use of virtual patients in medical school curricula. Advances in physiology education **36**(1), 48–53 (2012)
12. Coronado, E., Mastrogiovanni, F., Indurkhya, B., Venture, G.: Visual programming environments for end-user development of intelligent and social robots, a systematic review. Journal of Computer Languages **58**, 100970 (2020)
13. DeTore, N.R., Balogun-Mwangi, O., Eberlin, E.S., Dokholyan, K.N., Rizzo, A., Holt, D.J.: An artificial intelligence-based virtual human avatar application to assess the mental health of health care professionals: A validation study. Journal of Medical Extended Reality **1**(1), 215–226 (2024)
14. Duan, H., Wei, J., Wang, C., Liu, H., Fang, Y., Zhang, S., Lin, D., Chen, K.: Botchat: Evaluating llms' capabilities of having multi-turn dialogues. In: Findings of the association for computational linguistics: NAACL 2024. pp. 3184–3200 (2024)
15. Esposito, A., Calvano, M., Curci, A., Desolda, G., Lanzilotti, R., Lorusso, C., Piccinno, A.: End-user development for artificial intelligence: A systematic literature review. In: International Symposium on End User Development. pp. 19–34. Springer (2023)

16. Fischer, G., Giaccardi, E., Ye, Y., Sutcliffe, A.G., Mehandjiev, N.: Meta-design: a manifesto for end-user development. Communications of the ACM **47**(9), 33–37 (2004)
17. Fogli, D., Parasiliti Provenza, L.: A meta-design approach to the development of e-government services. Journal of Visual Languages amp; Computing **23**(2), 47–62 (Apr 2012). https://doi.org/10.1016/j.jvlc.2011.11.003
18. Ghosh, R., You, C., Venkatakrishnan, R., Lok, B.: Exploring the influence of multiple virtual agents for addressing barriers to cancer clinical trial participation. In: Proceedings of the 25th ACM International Conference on Intelligent Virtual Agents. pp. 1–10 (2025)
19. Gratch, J., Hartholt, A., Dehghani, M., Marsella, S.: Virtual humans: a new toolkit for cognitive science research. In: Proceedings of the Annual Meeting of the Cognitive Science Society. vol. 35 (2013)
20. Gris, I., Novick, D.G.: Virtual agent interaction framework (vaif): a tool for rapid development of social agents (2018)
21. Hartholt, A., Fast, E., Kim, K., Leeds, A., Sookiassian, E., Mozgai, S.: Multi-platform intelligent agents with the virtual human toolkit. In: Proceedings of the 25th ACM International Conference on Intelligent Virtual Agents. pp. 1–3 (2025)
22. Hartholt, A., Fast, E., Li, Z., Kim, K., Leeds, A., Mozgai, S.: Re-architecting the virtual human toolkit: towards an interoperable platform for embodied conversational agent research and development. In: Proceedings of the 22nd ACM International Conference on Intelligent Virtual Agents. pp. 1–8 (2022)
23. Hartholt, A., Traum, D., Marsella, S.C., Shapiro, A., Stratou, G., Leuski, A., Morency, L.P., Gratch, J.: All together now: Introducing the virtual human toolkit. In: Intelligent Virtual Agents: 13th International Conference, IVA 2013, Edinburgh, UK, August 29-31, 2013. Proceedings 13. pp. 368–381. Springer (2013)
24. He, X., Zhang, T., Pan, M., Ma, Z., Hu, C.J.: Template-based model generation. Software amp; Systems Modeling **18**(3), 2051–2092 (Nov 2017). https://doi.org/10.1007/s10270-017-0634-5
25. Huang, X., Ruan, W., Huang, W., Jin, G., Dong, Y., Wu, C., Bensalem, S., Mu, R., Qi, Y., Zhao, X., et al.: A survey of safety and trustworthiness of large language models through the lens of verification and validation. Artificial Intelligence Review **57**(7), 175 (2024)
26. Hussain, S., Ameri Sianaki, O., Ababneh, N.: A survey on conversational agents/chatbots classification and design techniques. In: Workshops of the International Conference on Advanced Information Networking and Applications. pp. 946–956. Springer (2019)
27. Jiang, Z., Huang, X., Wang, Z., Liu, Y., Huang, L., Luo, X.: Embodied conversational agents for chronic diseases: scoping review. Journal of Medical Internet Research **26**, e47134 (2024)
28. Kohli, G.S.: Evaluating large language models for conversational ai agents. Journal of Computational Analysis & Applications **34**(10) (2025)
29. Krieger, J.L., Neil, J.M., Duke, K.A., Zalake, M.S., Tavassoli, F., Vilaro, M.J., Wilson-Howard, D.S., Chavez, S.Y., Laber, E.B., Davidian, M., et al.: A pilot study examining the efficacy of delivering colorectal cancer screening messages via virtual health assistants. American journal of preventive medicine **61**(2), 251–255 (2021)
30. Laranjo, L., Dunn, A.G., Tong, H.L., Kocaballi, A.B., Chen, J., Bashir, R., Surian, D., Gallego, B., Magrabi, F., Lau, A.Y., et al.: Conversational agents in healthcare: a systematic review. Journal of the American Medical Informatics Association **25**(9), 1248–1258 (2018)

31. Lee, J., Kim, H., Kim, K.H., Jung, D., Jowsey, T., Webster, C.S.: Effective virtual patient simulators for medical communication training: a systematic review. Medical education **54**(9), 786–795 (2020)
32. Liaw, S.Y., Ooi, S.W., Rusli, K.D.B., Lau, T.C., Tam, W.W.S., Chua, W.L.: Nurse-physician communication team training in virtual reality versus live simulations: randomized controlled trial on team communication and teamwork attitudes. Journal of medical Internet research **22**(4), e17279 (2020)
33. Lieberman, H., Paternò, F., Klann, M., Wulf, V.: End-user development: An emerging paradigm. In: End user development, pp. 1–8. Springer (2006)
34. Lok, B., Foster, A.E.: Can virtual humans teach empathy? Teaching Empathy in Healthcare: Building a New Core Competency pp. 143–163 (2019)
35. Lucas, G.M., Rizzo, A., Gratch, J., Scherer, S., Stratou, G., Boberg, J., Morency, L.P.: Reporting mental health symptoms: breaking down barriers to care with virtual human interviewers. Frontiers in Robotics and AI **4**, 51 (2017)
36. MacNeill, A.L., MacNeill, L., Luke, A., Doucet, S.: Health professionals' views on the use of conversational agents for health care: qualitative descriptive study. Journal of Medical Internet Research **26**, e49387 (2024)
37. Micoulaud-Franchi, J.A., Sagaspe, P., De Sevin, E., Bioulac, S., Sauteraud, A., Philip, P.: Acceptability of embodied conversational agent in a health care context. In: International conference on intelligent virtual agents. pp. 416–419. Springer (2016)
38. Mollick, E.R., Mollick, L.: Using ai to implement effective teaching strategies in classrooms: Five strategies, including prompts. The Wharton School Research Paper (2023)
39. Myers, B., Hudson, S.E., Pausch, R.: Past, present, and future of user interface software tools. ACM Transactions on Computer-Human Interaction (TOCHI) **7**(1), 3–28 (2000)
40. Nouraei, F., Yong, Z., Bickmore, T.: Healthdial: A no-code llm-assisted dialogue authoring tool for healthcare virtual agents. arXiv preprint arXiv:2510.15898 (2025)
41. Park, J., Choo, S.: Generative ai prompt engineering for educators: Practical strategies. Journal of Special Education Technology **40**(3), 411–417 (2025)
42. Paternò, F.: End user development: Survey of an emerging field for empowering people. International Scholarly Research Notices **2013**(1), 532659 (2013)
43. Peng, X., Koch, J., Mackay, W.E.: Designprompt: Using multimodal interaction for design exploration with generative ai. In: Proceedings of the 2024 ACM Designing Interactive Systems Conference. pp. 804–818 (2024)
44. Piro, L., Desolda, G., Matera, M., Lanzilotti, R., Mosca, S., Pucci, E.: An interactive paradigm for the end-user development of chatbots for data exploration. In: IFIP Conference on Human-Computer Interaction. pp. 177–186. Springer (2021)
45. Repenning, A., Ioannidou, A.: Agent-based end-user development. Communications of the ACM **47**(9), 43–46 (Sep 2004). https://doi.org/10.1145/1015864.1015887
46. Repenning, A., Ioannidou, A.: What makes end-user development tick? 13 design guidelines. In: End user development, pp. 51–85. Springer (2006)
47. Rizzo, A., Lange, B., Buckwalter, J.G., Forbell, E., Kim, J., Sagae, K., Williams, J., Difede, J., Rothbaum, B.O., Reger, G., Parsons, T., Kenny, P.: Simcoach: an intelligent virtual human system for providing healthcare information and support. International Journal on Disability and Human Development **10**(4), 277–281 (2011). https://doi.org/doi:10.1515/IJDHD.2011.046, https://doi.org/10.1515/IJDHD.2011.046

48. Rossen, B., Lok, B.: A crowdsourcing method to develop virtual human conversational agents. International Journal of Human-Computer Studies **70**(4), 301–319 (2012)
49. Rough, D., Cowan, B.: Poster: Apis for ipas? towards end-user tailoring of intelligent personal assistants. In: 2020 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC). pp. 1–2. IEEE (2020)
50. Schoenthaler, A., Albright, G., Hibbard, J., Goldman, R., et al.: Simulated conversations with virtual humans to improve patient-provider communication and reduce unnecessary prescriptions for antibiotics: a repeated measure pilot study. JMIR medical education **3**(1), e6305 (2017)
51. Vilaro, M.J., McAlindon, K., Mertens, G., Ashley, T., Zalake, M., Cooks, E.J., Krieger, J.L.: Information architects: using community-engaged and qualitative methods to design a technology-based nutrition and cancer risk intervention for rural adults. Cancer Control **30**, 10732748221130162 (2023)
52. Vilaro, M.J., Wilson-Howard, D.S., Zalake, M.S., Tavassoli, F., Lok, B.C., Modave, F.P., George, T.J., Odedina, F., Carek, P.J., Krieger, J.L.: Key changes to improve social presence of a virtual health assistant promoting colorectal cancer screening informed by a technology acceptance model. BMC medical informatics and decision making **21**(1), 196 (2021)
53. Yao, H., Gomes de Siqueira, A., Rogers, M.L., Bloch-Elkouby, S., Lawrence, O., Sarli, G., Foster, A., Mitelman, S.A., Galynker, I., Lok, B.: The impact of scaffolded and non-scaffolded suicidal virtual human interaction training on clinician emotional self-awareness, empathic communication, and clinical efficacy. BMC medical education **24**(1), 413 (2024)
54. You, C., Ghosh, R., Vilaro, M., Venkatakrishnan, R., Venkatakrishnan, R., Maxim, A., Peng, X., Tamboli, D., Lok, B.: Alter egos alter engagement: perspective-taking can improve disclosure quantity and depth to ai chatbots in promoting mental wellbeing. Frontiers in Digital Health **7**, 1655860 (2025)
55. Zalake, M.: Designing Effective Virtual Human Conversations to Promote Healthy Behaviors. Ph.D. thesis, University of Florida (2022)
56. Zalake, M., de Siqueira, A.G., Vaddiparti, K., Antonenko, P., Hamza-Lup, F., Lok, B.: Towards rapid development of conversational virtual humans using web3d technologies. In: Proceedings of the 25th International Conference on 3D Web Technology. pp. 1–2 (2020)
57. Zalake, M., Tavassoli, F., Griffin, L., Krieger, J., Lok, B.: Internet-based tailored virtual human health intervention to promote colorectal cancer screening: design guidelines from two user studies. In: Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents. pp. 73–80 (2019)
58. Zamfirescu-Pereira, J.D., Wong, R.Y., Hartmann, B., Yang, Q.: Why johnny can't prompt: how non-ai experts try (and fail) to design llm prompts. In: Proceedings of the 2023 CHI conference on human factors in computing systems. pp. 1–21 (2023)