Title

Toward a Unified Metadata Schema for Ecological Momentary Assessment with Voice-First Virtual Assistants.

Authors

Chen, Chen
Mrini, Khalil
Charles, Kemeberly
et al.

Publication Date

2021-07-01

DOI

10.1145/3469595.3469626

Peer reviewed

Published in final edited form as:

Proc 3rd Conf Conversat User Interfaces CUI 2021 (2021). 2021 July ; 2021: .

doi:10.1145/3469595.3469626.

Toward a Unified Metadata Schema for Ecological Momentary Assessment with Voice-First Virtual Assistants

Chen Chen,

Computer Science and Engineering, University of California San Diego, La Jolla, CA, USA

Khalil Mrini,

Computer Science and Engineering, University of California San Diego, La Jolla, CA, USA

Kemeberly Charles,

School of Medicine, University of California San Diego, La Jolla, CA, USA

Ella T. Lifset,

Biological Sciences, University of California San Diego, La Jolla, CA, USA

Michael Hogarth,

School of Medicine, University of California San Diego, La Jolla, CA, USA

Alison A. Moore,

School of Medicine, University of California San Diego, La Jolla, CA, USA

Nadir Weibel,

Computer Science and Engineering, University of California San Diego, La Jolla, CA, USA

Emilia Farcas

Qualcomm Institute, University of California San Diego, La Jolla, CA, USA

Abstract

Ecological momentary assessment (EMA) is used to evaluate subjects' behaviors and moods in their natural environments, yet collecting real-time and self-report data with EMA is challenging due to user burden. Integrating voice into EMA data collection platforms through today's intelligent virtual assistants (IVAs) is promising due to hands-free and eye-free nature. However, efficiently managing conversations and EMAs is non-trivial and time consuming due to the ambiguity of the voice input. We approach this problem by rethinking the data modeling of EMA questions and what is needed to deploy them on voice-first user interfaces. We propose a unified metadata schema that models EMA questions and the necessary attributes to effectively and efficiently integrate voice as a new EMA modality. Our schema allows user experience researchers to write simple rules that can be rendered at run-time, instead of having to edit the source code. We showcase an example EMA survey implemented with our schema, which can run on multiple voice-only and voice-first devices. We believe that our work will accelerate the iterative prototyping and design process of real-world voice-based EMA data collection platforms.

chenchen@ucsd.edu .

Keywords

Voice Assistant; Voice First Interface; Healthcare; Ecological Momentary Assessment; Data Modelling

1 INTRODUCTION

Ecological momentary assessment (EMA) is an important technique in behavioral science to collect *in situ* research participants' behaviors, experiences, and moods in their natural setting [19]. Collecting real-time and temporally-dense participants' self-report EMA data in an ecologically valid setting is valuable, yet challenging, especially for under-represented groups (*e.g.,* older adults and people with physical or mental disabilities) [8, 14]. The design of EMA data collection platforms must consider various interconnected concerns, such as user engagement, reporting burden, data validity, and honest disclosure [7].

Over the last few years, smartwatches have been considered as an effective tool to support EMA. For example, Intille *et al.* [10, 15, 16] designed the μEMA on smartwatch using a 1-tap glance-able microinteraction to collect participants' mood states to understand the instantaneous mood that participants felt while getting notifications. They found that μEMA can reduce the perceived cognitive burden and device access time, and thus increase the EMA response rate [10, 15].

Integrating *voice interfaces* into EMA platforms also promises to increase user engagement, as participants would benefit from the hands-free and eye-free nature of voice-based devices. Using those devices can also reduce the device access time (*i.e.,* time that participants take to access the device and start the EMA questionnaire) and usage time (*i.e.,* time for participants to complete the assigned list of EMA questions). Researchers also proposed the *voice-first* design [3] attempting to incorporate an additional visual modality allowing users to interact through a built-in touch screen. This shift is also visible in today's intelligent virtual assistants (IVAs) that have been built into various voice-enabled smart devices and often include additional displays and touch interfaces (*e.g.,* Echo Show[1]).

Designing a usable conversational system for EMA data collection on such platforms is challenging. When using methods such as rapid prototyping and iterative design, Human-Computer Interaction (HCI) and User Experience (UX) researchers need to carefully consider how to design both the information output (*e.g.,* how to announce EMA questions in a correct form and prompt users upon failures?) and information input (*e.g.,* what are the users' possible intents?) [9, 18]. Prototyping such EMA systems on IVAs, while carefully considering those questions, is non-trivial and time consuming, and currently hinders the design process. To better understand the hurdles of deploying EMAs on voice-based IVAs, we break down the design process into three core challenges.

[1]Amazon Echo Show: https://www.amazon.com/Echo-Show-8/dp/B07PF1Y28C

First, current commercial systems are not designed with the concept of a prototype in mind. To evaluate the usability of a conversational voice user interface, complete systems need to be developed in complex IVA environments geared to develop and deploy products, which often means losing control on the application itself. To maintain more control, existing works attempted to build hardware-software systems from scratch, such as speech systems using a Raspberry Pi board for conducting Wizard of Oz studies [4]. However, it is difficult for these custom-based solutions to scale to large research projects, and it is not feasible to replicate multiple such prototypes for large populations. In addition, the recording of private speech (*i.e.,* conversations outside of the EMA research questions) on the vendor's cloud infrastructure is problematic, as it could potentially break multiple ethical research regulations. To address this privacy issue, MicShield [20] proposed using an additional mechanism to ensure that the privacy of unintended speech is preserved, using inaudible ultrasound signals that mask conversations not directed to the voice-based system. However, such approaches would incur additional prototyping time and effort. Similar to the example above, these cutting-edge research approaches are far from being integratable in systems that can actually be deployed with research subjects, leading to issues of scalability and accessibility.

Second, although commercially-available voice survey platforms (*e.g.,* SurveyLine[2]) provide a conversational speech system for potentially collecting EMA data, such platforms only offer standard services that are difficult to customize to the needs of a particular study. For example, existing platforms do not allow researchers to configure the occurrence of EMA questions based on context (*e.g.,* time of the day) in an easy and flexible way. Furthermore, depending on the type of data being collected, studies may be subject to local data privacy or health laws such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States, which further limit the third-party services that can be used for research.

Third, a practical voice-based EMA data collection platform is beyond a simple sequential question and answering interface and it usually requires more advanced, yet essential features, such as conditional branching and combining input from different type of responses [24].

Although commercially-available voice assistant vendors provide rule-based intent design approaches for developers to fast prototype voice apps using serverless functions, the stateless nature of these solutions makes it difficult to track conversation flows. Graphical programming methods for defining the conversation flows (*e.g.,* VoiceFlow[3] and kore.ai[4]) are geared to solve this problem, but currently do not offer enough flexibility and have important scalability issues.

In this work, we propose the design of a metadata schema and programming model to support healthcare and behavioral science researchers to rapidly prototype a practical EMA

[2]SurveyLine: https://www.surveysbyvoice.com
[3]VoiceFlow: https://www.voiceflow.com
[4]Kore.ai: https://kore.ai

data collection system that can be easily provisioned on voice-first digital assistants. Instead of advocating for the design of a customized hardware-software system from scratch, we leverage commercially available hardware that is affordable and easy to set-up.

To the end, we implement an EMA data collection platform using our proposed metadata schema on top of Amazon-based voice-first digital assistants (Amazon Echo Dot, Amazon Echo Show, and Alexa assistant running on top of iPhone and Apple Watch) with Amazon DynamoDB as the back-end data storage. Although we are using Alexa assistants as the running example, our proposed schema can be transferred and generalized to other commercially available voice-first digital assistants.

2 SCHEMA DESIGN

With the Alexa ecosystem as an example, we consider an end-to-end system like the one shown in Figure 1, where the user's speech is captured through various devices with voice-based IVAs. The Alexa Voice Service (AVS) is used to process and understand the user's speech. An Alexa skill and lambda function are used to parse and forward the transcribed text, and to receive the EMA questions with graphic and/or audio elements to be rendered or announced through the front-end hardware.

Our proposed schema is used to define how a database stores the EMA questionnaires and how the back-end infrastructure manages the conversation flow. Notably, during the iterative design process, researchers only need to change the meta attributes stored in the database and do not have to engage with source code. We envision the creation of a web-based interface for researchers to specify the attributes.

Designing a schema to support prototyping EMAs on top of today's voice-first IVAs is non-trivial due to the diverse possibilities of users' intents, the requirement of controlling the occurrence of EMA questions based on "real world" contexts [3], and the complexities of additional modalities provided by voice-first interfaces (*e.g.,* the visual output and touch input). In this section, we describe the design and implementations of primitive building components.

2.1 Entity Relationships and Cross-Platform Support

Figure 2 shows the entity-relationships diagram of our proposed schema. We model each entity as a separate collection. Relations connect different entities of the voice-first EMA infrastructure. One of our target groups are UX researchers who aim to rapidly prototype a voice-based EMA platform for evaluating usability, without needing to build state machines for managing conversation flows. Therefore, we designed our model to be minimal, intuitive, and flexible enough for them to tune and reconfigure the system. We now describe the main entities in our model:

• **EMA_Topic** and **EMA_Question_Node** encapsulate the EMA questions. Multiple **root_questions** can be defined as part of **EMA_Topic**. This is where the occurrence of each EMA question would be triggered based on the user-defined context. Also, multiple

paraphrased questions can be provided for the same topic, and the system can pick randomly between them to reduce user boredom.

• Since our schema is designed for different types of multi-modal devices, we introduce **Visual_Output** and **Audio_Output** as two separated collections shareable by multiple **EMA_Question_Nodes**. For example, multiple EMA questions with a 5-point Likert scale input might share the same **Visual_Output** and define five buttons when deployed on voice-first devices with touchable input. Notably, multiple audio scripts can be pre-defined as part of the **Audio_Output** entity, which aims to provide users a feeling of conversation, potentially enhancing the user engagement [21]. The **Visual_Output** collection defines basic properties for instructing voice-first devices to render the graphic interactive widgets (*e.g.,* buttons, sliders, *etc.*).

• Our schema also defines the concept of **Answer**, which provides a way for practitioners to define the correct rule for validating users' responses and providing feedback to users (*e.g.,* prompting error messages) (see Sec. 2.4). We specify the **number_of_attempts** as part of the **Answer** collection as the maximum number of times that participants may correct their previous responses.

• Each **EMA_Question_Node** can connect to a number of different **EMA_Question_Conditions**, indicating the possible transitions between conversations. Also, each **Schedule** can be shared by multiple **EMA_Question_Nodes**, where the occurrences of each question can be determined by the different time contexts.

2.2 Contextual Awareness and Conditions

Contextual awareness requires the occurrence of specific EMA questions, depending on predefined real-world conditions (*i.e.,* the context). For example, such context can include a particular time, the weather obtained from a remote weather services, previous answers, and sensor data. Our model allows the flow of the EMA conversation to largely depend on such contextual information. Due to the context being highly heterogeneous, our schema includes a method for researchers to easily define contextual rules during initial prototypes.

Inspired by the *function* component available in many programming languages, we incorporated a **condition** property that includes both a **data_fetching_rule** and **return_rule** as part of the **EMA_Quest ion_Condition**. The **data_fetching_rule** field contains code defining rules for either fetching remote data or processing input data; the **return_rule** field contains code defining the condition to evaluate the input from the data fetching.

We now use *conditional branching* as a running example to describe how our model supports contextual awareness, specifically how a participant's previous EMA response is used as the context to decide the next EMA question. The use of conditional branching (*a.k.a.,* skip logic) is a well-known practice in designing questionnaires, where the respondents receives a different question based on how they answer the current question.

Existing work [17] has shown the benefits of using the principle of micro-interaction, which aims to reduce device access-time and usage-time with the goal of decreasing completion time, dropout rate, and support more accurate data entry. It has been demonstrated that this method is effective specifically in increasing compliance rate and completion rate if applied to the design of EMA data collection platforms on smartwatches [17]. We think that applying this approach also to voice-based IVAs, and thus grounding our model on it, would result in similar increased efficiency.

In particular, our schema models each EMA questionnaire as a decision tree, where questions and conditions are modelled as nodes and edges of the tree (see Fig. 3a). The **EMA_Question_Condition** entity contains the ID of two connected **EMA_Question_Nodes**: **prev_ema_question_node_id** and **next_ema_question_node_id** (see Fig. 3b and 3c). Figure 3 also illustrates how we model an example question when the response from the previous question is used as the context. Notably, researchers can simply treat the **_answer_** property as a built-in variable that stores the response from users. When the response is stored, then a rule defining the success or failure of a particular condition can be devised based on the user input, which can be therefore used to determine the subsequent EMA question based on the previous response.

Figure 3d shows another example where researchers can define a remote procedure call (RPC) to collect data (*e.g.,* current temperature) from a third-party service. At run time, we use **fork()** and **exec()** as techniques to spawn other processes to execute the rule defined, if this is required. Our current prototype uses the spawned process as an additional sandbox to execute the rules. However, in the long term, alternative isolated sandbox (*e.g.,* containers and remote instances) might be considered to address scalability and security challenges.

2.3 Schedule and Occurrence

Often the EMA question should occur at a specific time, which can be defined by the researchers in our meta-model as well. To do that, our schema realizes and natively integrates the concepts of *schedule* and *occurrence*. The **Schedule** entity defines the range of time when each question is scheduled to be prompted each day. In order to model the concept of occurrence, we take into consideration the timestamp of previous attempts of the same question, which is cached on the back-end. For example, if we define **occurrence** as 3600 seconds and **max_number_of_occurrence** as 2, this means that in any one hour interval, the same question can be prompted no more than twice.

2.4 Answer Validation and Error Prompt

Compared to approaches that use a Graphical User Interface (GUI) with touch and mouse-click based inputs, a key challenge that voice-first devices are facing is the ambiguity and unpredictability of users' inputs [6]. This means that EMA systems based on voice-driven conversations need a way to provide explicit error-recovery guidance. However, due to the variety of errors that can occur during the execution of voice-based applications, it is impractical to provide appropriate error messages at prototyping time. This means that the design of error messages can only be achieved after multiple rounds of the iterative process.

To address this problem, we included in our answer validation schema specific ways to define error prompting messages during the EMA testing phase.

Similar to context descriptions (see Sec. 2.2), these error message rules are rendered at run time using *reflection* techniques. Instead of requiring researchers to revisit the source code, only little effort is required to write these rules. Figure 4a shows an example **Answer** entity and how it can be used to validate a typical 5-point Likert scale input. As part of future work, our schema could also support researchers to use RPC, where a remote cloud service such as [23] can check the correctness of participants' responses in real-time using smarter natural language understanding models, and thus generate corresponding error messages (Fig. 4b).

3 EXAMPLE APPLICATIONS

To better showcase our proposed schema we outline how to rapid prototype an example EMA questionnaire with conditional branching properties. We reused the EMA questionnaire designed by Maher *et al.* [12], to evaluate the sedentary behavior and physical activity of older adults (see Fig. 5) and deployed it on Amazon Alexa IVAs, using DynamoDB to implement our data model. We built an end-to-end system as shown in Figure 1 by integrating an Alexa skill, lambda functions, and a python flask server.[5] We used the Alexa Presentation Language (APL) to define the attributes of the visual elements [2].

Figure 6 shows our example application deployed on different *user-detached* voice-first devices—usually standalone and affiliated to a particular fixed environment—and *user-attached* voice-first devices—typically carried or worn by users—using our proposed metadata schema. Figure 6a and 6b show the prototype running on Amazon Echo Dot, which is user-detached and only supports speech input. Figure 6c, 6d and 6e show the prototype running on Amazon Echo Show, a popular user-detached voice-first user interface with built-in touch screen. With our schema, researchers were able to easily define a number of input widgets beyond voice, using touchable buttons (see Fig. 6e). Similarly, our prototype can also run on user-attached mobile devices. We showcase our example using the Alexa App running on iPhone 12 Pro (see Fig 6f and - 6g). As the current Alexa app is not supported by the Apple Watch Series 3, we use the *Voice in a Can*[6] framework to receive, process, and render the interaction process (see Figs 6h and 6i). Notably, in order to modify the questions, answer types, visual and audio outputs, and the conversation flow, researchers only need to refine the rules and metadata in the database, instead of revisiting any of the source code.

As this paper focuses on the design of a schema to assist rapid proof-of-concept prototypes, we leave the analysis of the effectiveness of different user interfaces and hardware embodiment for future work.

[5]Python Flask: https://flask.palletsprojects.com
[6]Voice in a Can: https://voiceinacan.com

4 CONCLUSION

We designed a novel unified metadata schema to enable Human-Computer Interaction (HCI), user experience (UX), and behavioral health researchers to rapidly prototype Ecological Momentary Assessment (EMA) data collection applications on different voice-first smart devices with built-in IVAs. The proposed schema supports critical features to enable the deployment of effective EMA surveys, and addresses multiple considerations and challenges introduced by the voice modality. We showcased an EMA implementation example that assess the physical activities and sedentary behaviors for older adults using user-attached and user-detached Amazon Alexa enabled devices.

We believe that our work will accelerate the design and prototyping process of voice-integrated EMA data collection platforms, and will ultimately enable voice-based IVAs to be used as a support for EMA data collections.

ACKNOWLEDGMENTS

This work is part of early effort of Project VOLI[7] [5, 11, 13, 22], and was supported by NIH/NIA under grant R56AG067393. We appreciate insightful feedback from the anonymous reviewers and fellow colleagues at the University of California San Diego.

REFERENCES

[1]. Amazon. 2021. Data Type Supported by Amazon DynamoDB. https://docs.aws.amazon.com/amazondynamodb/latest/APIReference/API_Types_Amazon_DynamoDB.html

[2]. Amazon APL. 2021. Understand Alexa Presentation Language (APL). https://developer.amazon.com/en-US/docs/alexa/alexa-presentation-language/understand-apl.html

[3]. Bajorek Joan Palmiter. 2018. Voice First Versus the Multimodal User Interfaces of the Future. https://www.uxmatters.com/mt/archives/2018/10/voice-first-versus-the-multimodal-user-interfaces-of-the-future.php

[4]. Brüggemeier Birgit and Lalone Philip. 2019. WoS - Open Source Wizard of Oz for Speech Systems. In IUI Workshops.

[5]. Chen Chen, Johnson Janet G., Charles Alice, Lee Kemeberly abd, Lifset Ella T., Hogarth Michael, Moore Alison A., Farcas Emilia, and Weibel Nadir. 2021. Understanding Barriers and Design Opportunities to Improve Healthcare and QOL for Older Adults through Voice Assistants. In The 23rd International ACM SIGACCESS Conference on Computers and Accessibility (Virtual Event, USA) (ASSETS '21). Association for Computing Machinery, Virtual Event, USA. 10.1145/3441852.3471218

[6]. Cohen Michael H., Giangola James P., and Balogh Jennifer. 2004. Voice User Interface Design. Addison-Wesley, Boston, MA.

[7]. Doherty Kevin, Balaskas Andreas, and Doherty Gavin. 2020. The Design of Ecological Momentary Assessment Technologies. 32, 3 (2020), 257–278.

[8]. Ferreira Denzil, Goncalves Jorge, Kostakos Vassilis, Barkhuus Louise, and Dey Anind K. 2014. Contextual Experience Sampling of Mobile Application Micro-Usage. In Proc. MobileHCI '14. 91–100.

[9]. Holland Jennifer. 2021. The Challenges & Advantages of Conversational User Interfaces in 2021. https://lform.com/blog/post/the-challenges-advantages-of-conversational-user-interfaces-in-2021/

[7]Prof. Michael Hogarth has an equity interest in LifeLink Inc. and also serves on the company's Scientific Advisory Board. The terms of this arrangement have been reviewed and approved by the UC San Diego in accordance with its conflict of interest policies.

[10]. Intille Stephen, Haynes Caitlin, Maniar Dharam, Ponnada Aditya, and Manjourides Justin. 2016. μEMA: Microinteraction-Based Ecological Momentary Assessment (EMA) Using a Smartwatch. In Proc. UbiComp '16. 1124–1128.

[11]. Johnson Janet, Mrini Khalil, Hogarth Michael, Moore Alison, Nakashole Nadpa, Weibel Nadir, and Farcas Emilia. 2020. Voice-Based Conversational Agents for Older Adults. In Adjunct Proc. CHI '20.

[12]. Maher Jaclyn P., Rebar Amanda L., and Dunton Genevieve F.. 2018. Ecological Momentary Assessment Is a Feasible and Valid Methodological Tool to Measure Older Adults' Physical Activity and Sedentary Behavior. Frontiers in Psychology 9 (2018), 1485. [PubMed: 30158891]

[13]. Mrini Khalil, Chen Chen, Nakashole Ndapa, Weibel Nadir, and Farcas Emilia. 2021. Medical Question Understanding and Answering for Older Adults. (2021). http://voli.ucsd.edu/pdfs/2021_VOLI_SoCal_NLP.pdf

[14]. Nagel Kristine S., Hudson James M., and Abowd Gregory D.. 2004. Predictors of Availability in Home Life Context-Mediated Communication. In Proc. CSCW '04. 497–506.

[15]. Ponnada Aditya, Haynes Caitlin, Maniar Dharam, Manjourides Justin, and Intille Stephen. 2017. Microinteraction Ecological Momentary Assessment Response Rates: Effect of Microinteractions or the Smartwatch? Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1, 3, Article 92 (2017). [PubMed: 30198012]

[16]. Ponnada Aditya, Binod Thapa-Chhetry Justin Manjourides, and Intille Stephen. 2021. Measuring Criterion Validity of Microinteraction Ecological Momentary Assessment (Micro-EMA): Exploratory Pilot Study With Physical Activity Measurement. JMIR Mhealth Uhealth 9, 3 (10 Mar 2021), e23391. [PubMed: 33688843]

[17]. QuestionPro. 2021. What is Survey Skip Logic and Branching? https://www.questionpro.com/features/branching.html

[18]. Sayago Sergio, Neves Barbara Barbosa, and Cowan Benjamin R. 2019. Voice Assistants and Older People: Some Open Issues. In Proc. CUI '19. Article 7.

[19]. Stone Arthur A and Shiftman Saul. 1994. Ecological Momentary Assessment (EMA) in Behavorial Medicine. Annals of Behavioral Medicine (1994).

[20]. Sun Ke, Chen Chen, and Zhang Xinyu. 2020. "Alexa, Stop Spying on Me!": Speech Privacy Protection against Voice Assistants. In Proc. SenSys '20. 298–311.

[21]. Survey Monkey. 2021. Make Surveys More Engaging When You Do These 5 Things. https://www.surveymonkey.com/curiosity/make-surveys-more-engaging-with-5-things/

[22]. UCSD VOLI. 2021. VOLI: Voice Assistant for Quality of Life and Healthcare Improvement in Aging Populations, http://voli.ucsd.edu

[23]. Wit.ai. 2021. Building Natural Language Experieces. https://wit.ai

[24]. Ziegler Josh. 2018. Communication is Hard - or How Our First Approach to Converastion Management Caused as Many Problems as It Solved. https://medium.com/navigating-the-conversation/communication-is-hard-part-2-1b7529398cc2

CCS CONCEPTS

- Human-centered computing → Systems and tools for interaction design.

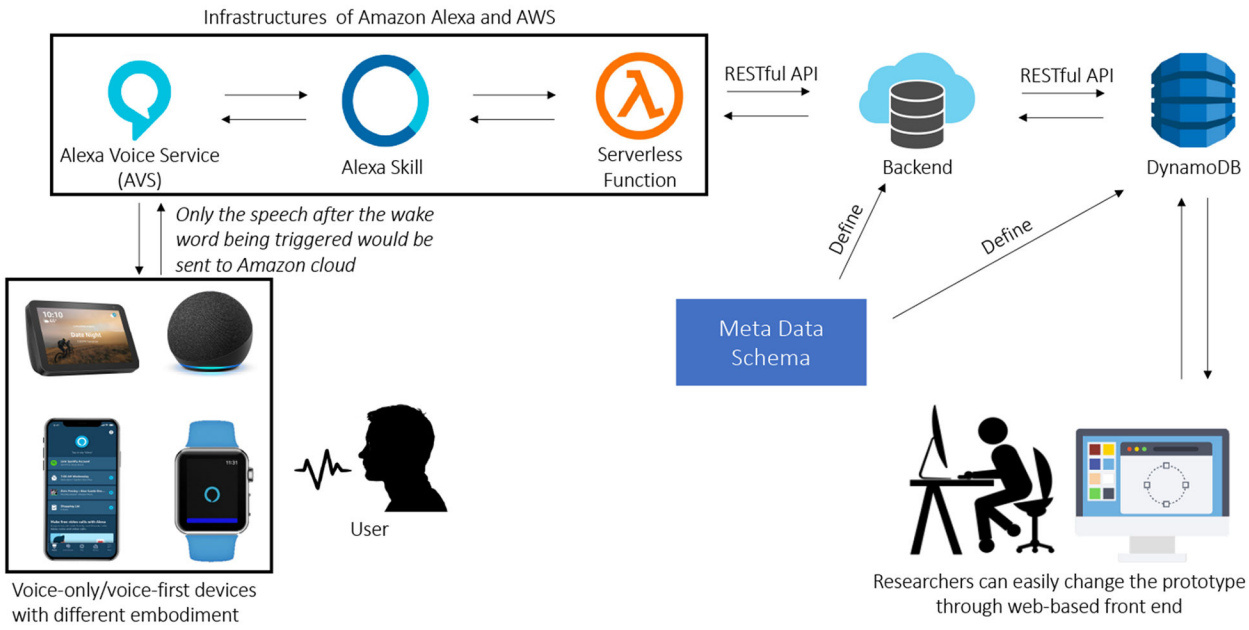

Figure 1:

Example system that uses our designed schema to store and render EMA questionnaires. To try out different conversation flows during the iterative design process, UX researchers only need to modify the content in the database.

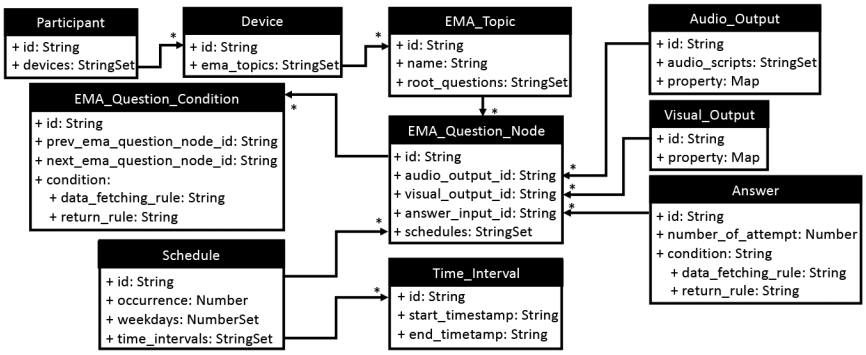

Figure 2:

A simplified entity relationships diagram for EMA questions and the necessary attributes. As a running example, we used the types supported by DynamoDB [1]. Notably, the *property* fields in *Audio_Output* and *Visual_Output* collections vary among types of questions.

Figure 3:

Modelling of conditional branching using a decision tree graph. (a) Decision tree modelling. (b) Example object of C1. (c) Example object of C2. (d) Example object with RPC.

a

```
{
  “id”: “1”,
  “number_of_attempt”: 5,
  “condition”: {
    “data_fetching”:
      “
      ret_msg = True
      if type(_answer_) != int:
            return \"Sorry! Your input type is invalid!\"
      elif _answer_ < 0:
            return \”Sorry! A valid input should not be negative!\”
      elif _answer_ > 4:
            return \”Sorry! A valid input should be larger than four!\"
      ”,
    “ret_condition”: “ret_msg”
  }
}
```

b

```
{
  “id”: “2”,
  “number_of_attempt”: 5,
  “condition”: {
    “data_fetching”:
      “
      def nlu_rpc(_input):
            # definition of nlu_rpc code
      err_msg = nlu_rpc(_answer_)
      if not err_msg:
            ret_msg = err_msg
      else:
            ret_msg = True
      ”,
    “ret_condition”: “ret_msg”
  }
}
```

Figure 4:

Example of how to instantiate an Answer in our metadata schema. (a) Example for validating 5–point Likert input. (b) Example using remote cloud services.

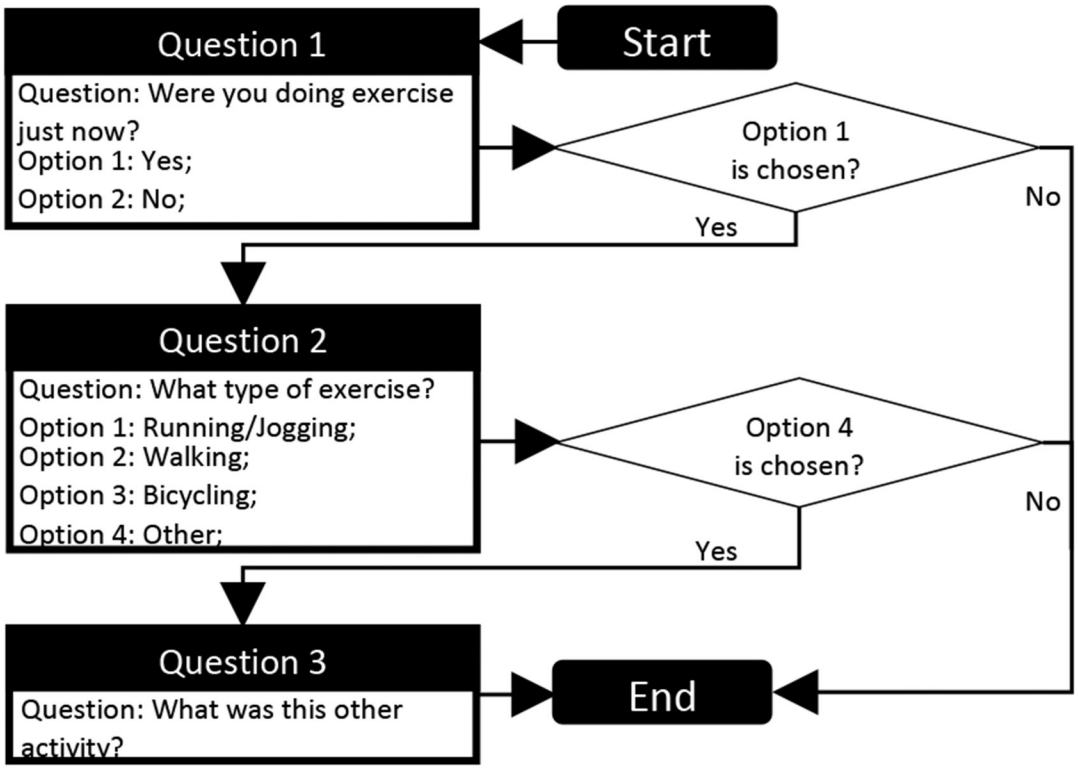

Figure 5:
Example EMA survey for evaluating sedentary behavior and physical activity revised from [12].

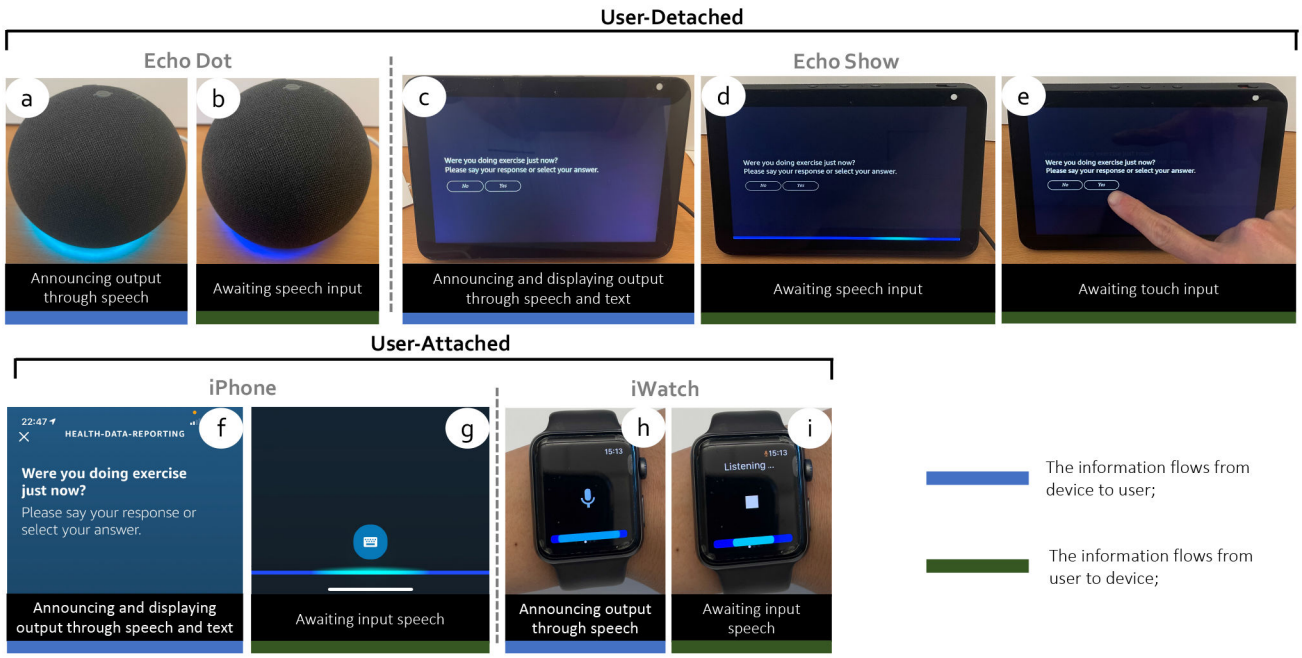

Figure 6:

Example EMA applications implemented on the standalone voice-only user interface realized by Amazon Echo Dot (a – b), the standalone voice-first user interface realized by Amazon Echo Show, where input can be achieved through speech (d) and touch (e), the voice-first interface realized by Amazon Alexa applications on iPhone (f – g), and voice-only user interface on wrist wearables, realized by Apple iWatch (h – i).