A Closed-Loop Approach for Improving the Wellness of Low-Income Elders at Home Using Game Consoles

Jong Hyun Lim, Andong Zhan, JeongGil Ko, Andreas Terzis, Sarah Szanton, and Laura Gitlin, Johns Hopkins University

ABSTRACT

As life expectancy in the industrialized world increases, so does the number of elders with chronic health conditions such as diabetes and congestive heart failure who require complex self-management routines. We present a motivational exercise gaming system whose goal is to increase the activity of elders with complex chronic conditions. Our gaming system, initially deployed unattended, showed discouraging results. Our second attempt addresses these shortcomings by coupling the gaming console with an application for presenting exercise results to remote clinicians and caregivers, and a smartphone-based application for collecting feedback and issuing alerts. HealthOS, a platform for developing healthcare applications, integrates all components of the applications.

INTRODUCTION

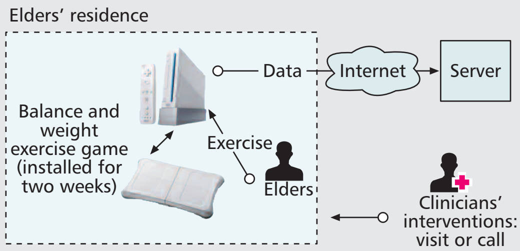
As life expectancy in the United States and around the world increases, the number of elders with complex chronic conditions impacting their physical function also rises. This number will continue to increase as the number of middle-aged adults reporting difficulty with physical function increases [1]. Moreover, most elders with disabilities prefer to live at home rather than nursing homes or other clinical facilities [2]. All these factors make the disability of such elders a major driver of healthcare costs and can also lead to poor quality of life, especially for those with low incomes. Solving these issues requires a low-cost longitudinal method of monitoring health and daily activities, as well as techniques for effective behavioral modifications.

To help such low-income elders, we developed a custom game for the Nintendo Wii to prevent fall risks through balance exercises and also to monitor the risk of heart attacks through long-term weight measurement. We started a pilot study with the Johns Hopkins University School of Nursing for six at-home elders (age ≥ 70) in Baltimore, Maryland. In doing so, we deployed gaming systems without any clinician attendance for two weeks after a day of initial training. We show the architecture of the deployed system on the left of Fig. 1. As the participating elders played the game, the measured data was transferred to the server for data analysis. In addition, nurses visited the participants according to their predefined schedules for periodic checkups. As part of the study, we conducted a set of Short Physical Performance Battery (SPPB) tests for the participants to see the difference in physical measures before and after the deployments. Afterward, we asked the participants to take a survey to quantify the difficulty of our gaming system for the elders. Our survey results indicate that the elders in our trial group had little or no difficulty in using the gaming system.

Disappointingly, as the right side of Fig. 1 shows, most participants only practiced the game 3.3 times on average out of 14 expected trials, leaving us with only a minimal amount of data to observe the system's clinical effectiveness. Even worse, some of the participants only practiced the game while the nurses were in attendance. This was so because elders with disabilities need persistent interventions to be encouraged, motivated, and assisted even when practicing an easy and motivational game. Furthermore, the functional loop formed by participants, gaming system, server, and clinicians was not closed, as visualization tools to present the collected data were not provided for real-time responses.

Although frequent home visits from nurses and calls through real-time monitoring tools can help alleviate the low utilization problem [3], they increase the cost of caregiving considerably. In addition, many elders with complex self-management needs have family members who live remotely [4]. These distant family members would like to be notified of medical alerts and can support elders with self-management tasks. Thus, the key question is: *How do we close the intervention loop so that we can minimize the caregivers' workload, include family members, and increase the quality of care at the same time*?

To answer this question, we redesigned our initial system by developing HealthOS, a cloud-based medical data management system, and DailyAlert, a server/smartphone application for



Participant	Actual trials/ expected trials (utilization)	Number of interventions (call/visit)
A	5/14(35%)	4
B	3/14(21%)	3
C	1/14(7%)	3
D	2/14(14%)	3
E	1/14(7%)	3
F	8/14(57%)	4

Figure 1. *Initial system architecture where we simply collected data from gaming systems for analyzing balance improvement (left). Low device utilization requires further interventions from clinicians/caregivers (right).*

alerting users. We present this new architecture in Fig. 2. Specifically, HealthOS collects data from gaming systems, publishes the data to various applications for visualization and analysis using Representational State Transfer (REST) application programming interfaces (APIs), and provides tools to add and manage new sensor devices. DailyAlert is a system in the cloud for clinicians and caregivers to build, schedule, and deliver timely alerts and information to elders' and their family members' smartphones. Moreover, DailyAlert analyzes the data pushed from HealthOS, and if certain conditions are met, it automatically generates alerts. We show that by using the two additional components in the monitoring process, we can form a fully closed-loop system that can improve and automate the intervention process of exercise monitoring for at-home elders. We believe that our approach can be used to improve the process of various applications in the healthcare domain where user interventions are necessary.

The remainder of this article is as follows. We discuss our proposed closed-loop approach in greater detail. We introduce our system's implementation, and conclude the article by introducing the potential impacts of our system.

Architecture

In this section, we describe each component in our proposed closed-loop approach (Fig. 2) in greater detail. We not only show the technical features of the components and their roles in the loop, but also discuss the benefits and impacts our system can have on elders, clinicians, caregivers, and family members, as well as system developers in pervasive healthcare scenarios.

Game Console for Elders

Gaming systems have received attention as an inexpensive motivational tool in pervasive healthcare scenarios. In this section, we summarize the experience obtained from our own pilot study and earlier work in healthcare applications of game systems. In doing so, we first categorize the findings into two major factors, *motivator* and *ability*, as suggested in Fogg's behavioral model [5].

Motivator: As earlier work suggests, it is beneficial to give pleasure and present the progress of a participant's exercise (e.g., through graphs) to motivate them in using the gaming system. In addition, our experiments showed that it is equally important to have visual/audio effects either during the game play or after completing a level (e.g., applause sounds or fireworks), to motivate elders to play the game.

Ability: An ability is defined as the most constraining factor for performing a particular task [5]. This constraining factor in our case is the elders' physical capabilities. Having considered this, Gerling *et al.* suggest developing easier games for elders with multiple levels of difficulties [6]. They also show that commercial games for Wii Fit or Wii Sports are not suitable for frail elderly players as they generally appear to be too complex and challenging for them. The other crucial factor is an *intuitive* and *user-friendly* software interface. For example, during our deployment, buttons placed side by side increased the chance of selecting the wrong option, which frustrated elders and reduced their willingness to play. Moreover, many of the elders in our study had difficulties selecting small icons on the screen as their hands were shaking while they were trying to hold the remote control still.

There have been previous attempts to develop motivating and user-friendly games for healthcare purposes. However, these games were used either in clinical settings [7] or during a clinician's visit at a residence, but were not deployed unattended [8]. However, as Fig. 1 suggests, the use of games in a long-term and standalone setting (i.e., without direct interventions from clinicians) may not be effective for elders. Rather, our deployments suggest that motivating and user-friendly games should be matched up with direct or remote interventions to further improve the effect of the game.

Considering their communication capability, game systems themselves have the potential to work as an interaction channel with caregivers. However, if voice chat is not supported, this intervention process may become a burden itself (e.g., typing text with game controllers was not easy for elders). Lyles *et al.* also indicated that some participants did not like the gaming systems for displaying the progress, results, or feedback [9]. Moreover, game systems are not ubiquitous (or mobile) devices that elders can use all the time for timely interaction.

To achieve ubiquitous intervention, earlier work has investigated using smartphones as part of the interaction loop [9]. Even though some subjects in the study reported difficulties in using the smartphones, they generally agreed that smartphones are advantageous for monitoring

Considering their communication capability, game systems themselves have the potential to work as an interaction channel with caregivers. However, if voice chat is not supported, this intervention process may become a burden itself.

martphones can be a channel that combines multiple interventions to a single device which simplifies the user interaction with the caregivers and clinicians. This suggests that our gaming system can further benefit from the use of smartphones.

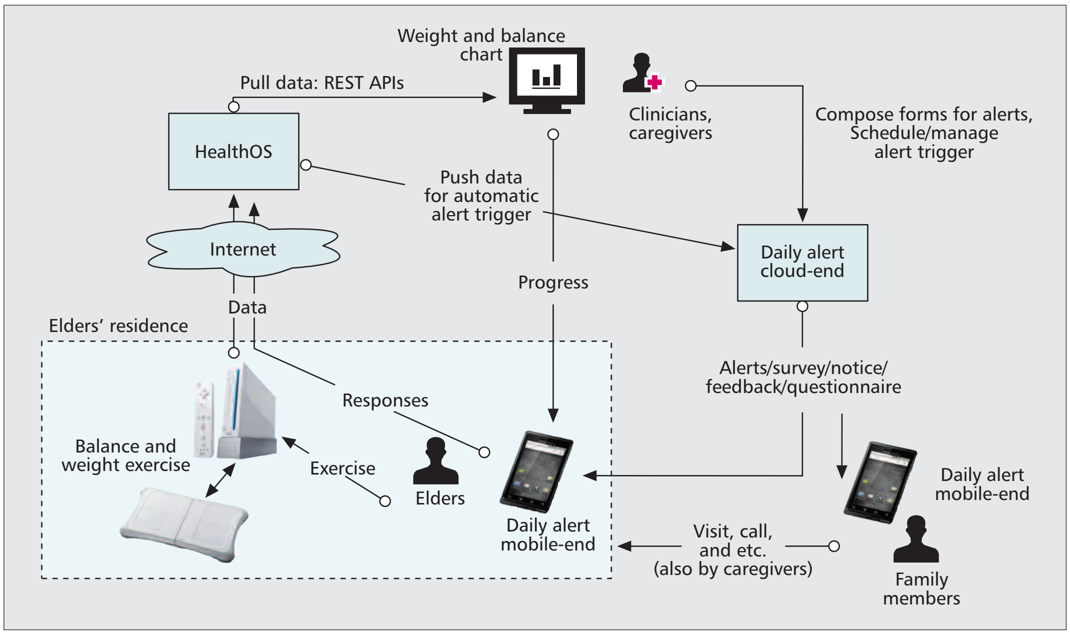


Figure 2. *Our implementation of a closed-loop approach using a game console for elders with HealthOS and DailyAlert. The goal is to create an environment facilitating intervention to elders economically and effectively. Thus, we implement our system with two extra components: HealthOS and DailyAlert with smartphones to 1) automate the intervention process, 2) provide elders with effective communication channels using easy interfaces, 3) include family members in the loop, and 4) provide developers with a simple development platform to extend the system with new sensing devices.*

their health. Additionally, smartphones can be a channel that combines multiple interventions to a single device which simplifies the user interaction with the caregivers and clinicians. This suggests that our gaming system can further benefit from the use of smartphones.

HealthOS: Data Collection and Development Platform

Most domestic healthcare applications use data collected from remote devices for further presentation to clinicians or caregivers. However, a majority of these applications are developed based on a "stovepipe" architecture [10]: no API is provided for retrieving data, heterogeneous message formats are used, and the devices are tightly coupled with the presentation through databases. This closed, vertically integrated nature impedes the composability and extendibility of the entire system as a whole. For instance, clinicians may need to combine the weight measurements from our gaming system with data from heart rate monitors to see the correlation between the two for detecting the chances of heart attacks. In doing so, if the heart rate monitor uses a different protocol, message format, and graphical user interface (GUI), integrating new devices to the application would require a substantial amount of time and effort.

To overcome such shortcomings, we introduce *HealthOS* into the intervention loop [11]. HealthOS is a middleware running in the cloud that collects data from pervasive healthcare devices, integrates healthcare applications, and also provides a development framework through RESTful APIs. Specifically, to accomplish these tasks, HealthOS consists of a set of *driver* and *pipeline* modules, as Fig. 3 shows. A driver module is a submodule running within the HealthOS framework and interfaces with diverse types of devices (e.g., different communication media and message formats). Once drivers collect and store the devices' data, they can be further translated through a set of pipeline modules to any medical record format that needs to be exposed.

The combination of drivers and pipelines provides three key benefits. First, as HealthOS collects data and translates it to the desired medical format, system developers may skip the translation phase and focus on implementing the presentation and analysis features of their applications. Second, since HealthOS accepts data from all devices and exposes the data with RESTful APIs, developers can easily combine data from multiple sensors; this opens up the possibility to develop innovative healthcare applications by combining multiple sensor readings, which have been impossible or difficult to do otherwise due to the isolated nature of the stovepipe architecture. Third, HealthOS can provide a unified framework to manage multiple device and access controls to multiple users. Particularly, the framework (e.g., web interfaces) enables adding new sensor devices.

Presentation and Analysis Applications in the Cloud

Systems in pervasive healthcare should not only collect data but also provide presentation tools (e.g., graphs and charts) for real-time monitoring and analysis. However, the tools in traditional systems are tightly coupled with their back-ends (e.g., databases). As a result, these tools are difficult to migrate to other platforms (e.g., smart-

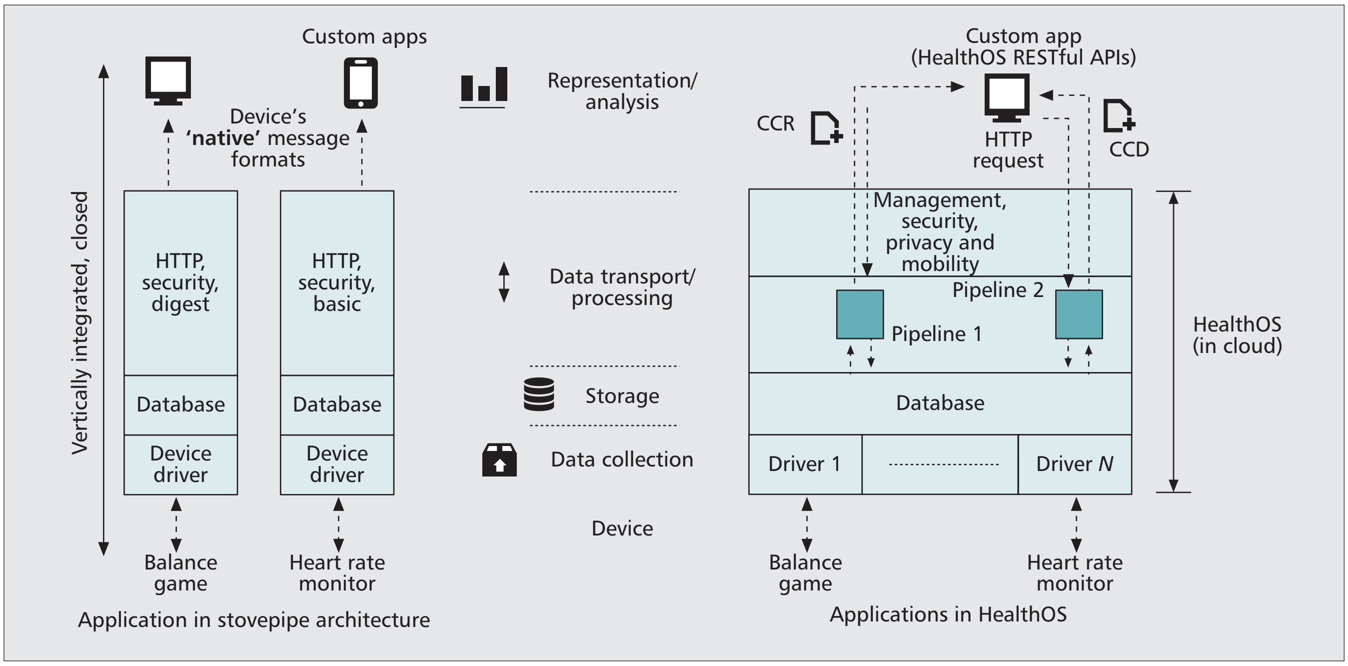


Figure 3. *In contrast to the vertically integrated, closed stovepipe architecture, HealthOS 1) collects sensor data through drivers, 2) translates the data to medical record formats such as ASTM Continuity of Care Record (CCR) through pipelines, 3) and provides RESTful interfaces to access the data and develop applications. Thus, by introducing HealthOS into the loop, new devices can easily be incorporated, and application developers can integrate the results from those heterogeneous devices.*

phones and handheld tablets for ubiquitous healthcare). Moreover, when tightly coupled, changing or modifying the back-ends can require developers to modify the presentation as well. Although services such as Open Database Connectivity can solve this problem by providing an abstraction layer between applications and database systems, they are not universally available on all platforms (e.g., mobile environments). Our approach, on the other hand, decouples the presentation and analysis front-end from the back-ends through a uniform interface.

In doing so, we suggest developing the presentation layer using HTTP and RESTful interfaces in the cloud. REST is a design paradigm to build and expose the resources that underlie a service [12]. In case of the web, clients wanting to access and retrieve resources mostly use URLs. For example, one can access patient A's weight data through an HTTP GET request to http://www.example.com/patientA/weight. This model implies that the presentation layer does not need to be tied to the semantics of services' back-ends. Doing so trivializes the data retrieval phase and simplifies transplanting applications to other platforms.

In our closed-loop architecture, HealthOS provides data to the application's presentation layer using HTTP and RESTful APIs. However, this does not mean that HealthOS must be the sole data repository applications use. Rather, presentation applications can also combine data from other web services in the Internet, such as Microsoft Health Vault. By combining data from multiple sources, the data presentation can become a more effective monitoring and analysis tool for caregivers and clinicians, while reducing development cost.

DailyAlert: Closing the Loop Using Mobile Intervention

Although increasing the number of human interventions (e.g., number of visits and calls) may have helped increase the device utilization in our initial system (Fig. 1), such human interventions can potentially increase the cost of caregiving. For example, placing phone calls to each elder may require a considerable number of man-hours. To address this issue, recent work has introduced the use of mobile phones and proved its potential to improve the wellness of elders, while decreasing the caregivers' workload (e.g., using email and text messaging) [9]. However, we believe that there is still room for improving mobile intervention.

Automation: Automating the intervention process can significantly reduce the workload of caregivers and clinicians. Specifically, the automated system should not only send periodic alerts, but also examine the collected data from the elders for alert generation. Once a decision is made, the system itself should automatically determine whether a new alert is needed.

Interface to build and schedule alerts: To make the alerts more effective, interfaces to build and control these alerts are essential. The clinicians or caregivers should be able to create new alerts, modify the context of the alert, and set up the schedule or constraints of the alert as well.

User-friendly software for elders: On the other hand, from the elders' perspective, the mobile application should be designed to provide a simple and intuitive interface. For example, it was helpful to minimize the amount of typing by replacing it with button clicks, and present pictures and voice-based messages instead

The combination of AlertTrigger and AlertFormat does not replace existing intervention methods. For example, the elders may need multiple visits to be retrained in how to use the devices. Rather, DailyAlert augments the number of interaction channels necessary to close the loop.

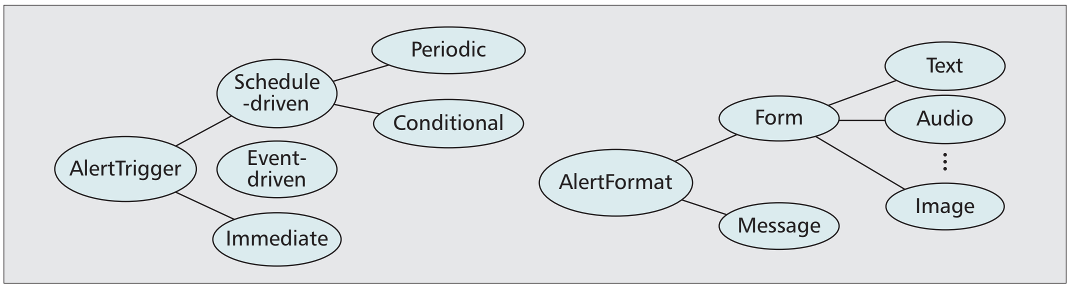


Figure 4. *An alert defined in DailyAlert system. Here, AlertTrigger defines when to trigger, and AlertFormat specifies the trigger's format. By supporting multiple temporal factors and formats, DailyAlert helps reducing caregivers' workload and provides elders with user-friendly software interfaces.*

of text in some cases, not to mention larger text size.

Having considered these remarks, we design a mobile intervention system called DailyAlert within the closed loop to complement existing intervention methods [13]. In DailyAlert, an alert is defined using two factors: when and how. Accordingly, as in Fig. 4, we design two system components, *AlertTrigger* and *AlertFormat*. Specifically, AlertTrigger can be categorized into three types: schedule-driven, event-driven, and immediate. An immediate trigger is used when caregivers want to transmit a one-time non-scheduled instant alert. When using a schedule-driven trigger, caregivers can set up periodic triggers, such as daily, weekly, monthly, or every *N* days with the specified time interval. In addition, a schedule-driven trigger can be combined with certain conditions. For example, when we want to alert the elders if there is no weight measurement until 6 p.m. each day, caregivers can specify a constraint, such as isWeighedToday(at 6PM) = false. DailyAlert then checks the constraint at 6 p.m. to see if the constraint is met and conditionally sends an alert.

One can also specify event-driven triggers. For example, one can setup an alert by describing sendAlert if weight ≥ 200 lbs. To do so, DailyAlert interacts with the cloud-based storage. Specifically, it either fetches the data from the cloud, as does Microsoft Health Vault, or uses data pushed by HealthOS. DailyAlert then checks the conditions and triggers alerts if needed. Therefore, by interacting with systems in the cloud and supporting multiple types of alerts, DailyAlert can contribute in reducing the caregivers' workload.

DailyAlert supports two alert formats: *message* and *form*. Caregivers use messages to alert users with text (e.g., directing elders to take medicine); therefore, messages are similar to SMS. If necessary, elders can also reply to messages or send new messages to caregivers and clinicians.

A form is a composite type of document supporting text, image, audio, and video, especially for questionnaires or surveys (e.g., SPPB tests in our pilot study). DailyAlert provides caregivers with a web interface to compose a form by using Open Data Kit (ODK) [14], which generates a self-describing XML schema for the form's structure. DailyAlert then distributes/pushes the composed form to the elders' smartphones. The key benefit of having such XML formats is that compared to text messaging, it increases readability and controllability. For instance, when asking elders about their pain level, instead of asking them to type a certain number, a form containing pictures representing pain levels on which the elders can click can be more intuitive and simple. In addition, when using forms, caregivers can decide how the form is going to be rendered on smartphones. Specifically, caregivers may want to render the answers in questionnaires in buttons, check boxes, text boxes, or images to have elders simply click or touch the screen, which would increase the controllability.

We note that the combination of AlertTrigger and AlertFormat does not replace existing intervention methods. For example, elders may need multiple visits to be retrained in how to use the devices. Rather, DailyAlert augments the number of interaction channels necessary to close the loop.

IMPLEMENTATION

In this section, we present the current status of the implementations for our proposed closed-loop system. Although we developed the system components primarily for our at-home elder care scenario, they are not constrained to a single application. Rather, we envision that our system with HealthOS and DailyAlert will be used in other healthcare studies (e.g., diabetes management) that require continuous intervention to change lifestyles and improve subjects' wellness.

BALANCE AND WEIGHT EXERCISE ON WII

To test the feasibility of our proposal for low-income elders, we developed a balance and weight exercising program on the Nintendo Wii. Our program focuses specifically on preventing falls by improving lower body strength through a balance game and monitoring long-term changes in weight to further predict potential heart attacks.

Although Wii Sports and Wii Fit are commercial state-of-the-art programs for healthcare purposes, they do not have the capability to send measurements online; hence, they do not fit the model of pervasive healthcare applications. We used a third-party program called WiiBrew to develop and run custom programs on the Wii. Specifically, using WiiBrew, we can run programs developed using C and C++ libraries ported to Nintendo Wii. Moreover, we can connect to the WiFi network and exchange messages with HealthOS when using WiiBrew.

By using the capabilities of Wii and WiiBrew, our program uses the Wii Fit board to take balance and weight measurements, and report the results in real time. In the balance game, participants were asked to shift their weight on the board to a particular position on the screen and hold still for a certain amount of time. The game provides nine different positions to shift the participants' weight and two difficulty levels. In addition, during the game, participants were asked to measure their weight and answer simple survey questions (e.g., "how are you feeling?") to correlate the effect of their mental and physical states on exercise performance.

HEALTHOS

Our current version of HealthOS (1.0) is developed in Python 2.6 and uses Tornado for web servicing. As Tornado supports WSDL, our implementation can be translated to systems supporting Python and WSDL (e.g., Google Web Apps). We also provide a web interface to manage caregivers/clinicians (e.g., add, delete, and perform access control), build pipelines to translate and expose stored data through RESTful interfaces, and also provide drivers for interfacing with devices. As HealthOS encompasses plenty of pipeline and driver modules, we design them based on a component-based software architecture to achieve modularity within the system. For example, when building a pipeline for a user's weight, HealthOS combines authentication, security, and format translation modules from a pool of different modules. By following the same interface rules and connecting the components when necessary, our system achieves light coupling among components. In turn, this light coupling enhances extensibility and scalability.

HealthOS provides macro scripts to compose drivers for new devices. Users can simply describe the devices' message formats (size and type of each field or XML schema) and communication interfaces (raw socket, USB serial, and HTTP are currently supported); with this information, the macro generates a simple module to decode the incoming messages from the devices. HealthOS also provides similar macros for creating pipeline modules. We believe that the two mechanisms simplify the integration of new devices significantly.

HealthOS uses Cyphertext-Policy Attribute-based Encryption (CP-ABE) to implement access controls and encrypt stored data [15]. Specifically, in CP-ABE, records are encrypted with secret keys and associated with policies that describe who is allowed to decrypt them. Such policies are expressed as Boolean formulas that reference attributes included in the secret key used to encrypt the record. Receivers that possess the attributes necessary to make the Boolean formula evaluate to true can recover the key and decrypt the record. By using CP-ABE, HealthOS can distribute secret keys in a scalable manner.

ELDER MONITOR FOR CAREGIVERS

To present the results collected at HealthOS to caregivers and clinicians, we developed a visualization and management GUI. Specifically, we implemented a web client running on the Google AppSpot, which can host cloud-based web applications. After the authentication and decryption processes, the application presents user information and connected devices, and retrieves the measurements stored in HealthOS. At this point, the balance and weight information is presented on a chart developed using Google Charts. In addition to this, the GUI also presents the responses to the surveys and questionnaires delivered from DailyAlert's cloud-end to the smartphones.

To present the results collected at HealthOS to caregivers and clinicians, we developed a visualization and management GUI. Specifically, we implemented a web client running on the Google AppSpot, which can host cloud-based web applications.

DAILYALERT APPLICATION

As Fig. 5a shows, DailyAlert consists of two system components: the mobile-end and the cloud-end. The cloud-end, developed using Java on Google App Engine, provides caregivers and clinicians with web interfaces to manage users (e.g., add or delete), build forms, and set up alert schedules. By using these services, caregivers and clinicians can define (or modify) an alert (consisting of a form, a group of elders to receive the form, and a specific schedule), set up conditions for automatic alert generation, and then activate the newly created alert in the system. Once activated, elders who are registered in the DailyAlert system receive alerts to encourage exercise, feedback on their exercise results, and questionnaires regarding their current status. Similarly, family members who need to visit/call/text elders also receive scheduled or event-based alerts.

Alert delivery is performed in a push-pull manner. First the cloud-end pushes an alert notification to the mobile-end; then the mobile-end can pull required data from the cloud-end to trigger an alert on the mobile device. Figure 5a illustrates this process. It might be counterintuitive that the mobile device needs to pull the data to retrieve the alerts. In fact, this is an implementation limitation we face when using the Google Cloud To Device Messaging (C2DM) service. Google C2DM provides the service to push messages to an Android phone, which can be used in a cellular network that is an essential part of our intervention process. However, Google C2DM puts a 1 Mbyte limitation on the size of the message that can be pushed. To make alerts extensible (e.g., include images and even videos), we push a small message with the essential data (e.g., a URL) and allow the mobile device to actively fetch the alert.

After the mobile-end device retrieves the alert schedule and form definition from the cloud-end, it maintains an alert task list. Unless changes are made by caregivers on the cloud-end, the mobile-end shows the downloaded form based on the defined schedules. By doing so, even though mobile-ends get disconnected, the scheduled alert will be displayed at predefined times and can store the elder's response locally. Therefore, when connectivity is regained, the cached response can be delivered to the designated storage services.

SUMMARY AND OUTLOOK

In this article, we propose the use of a closed-

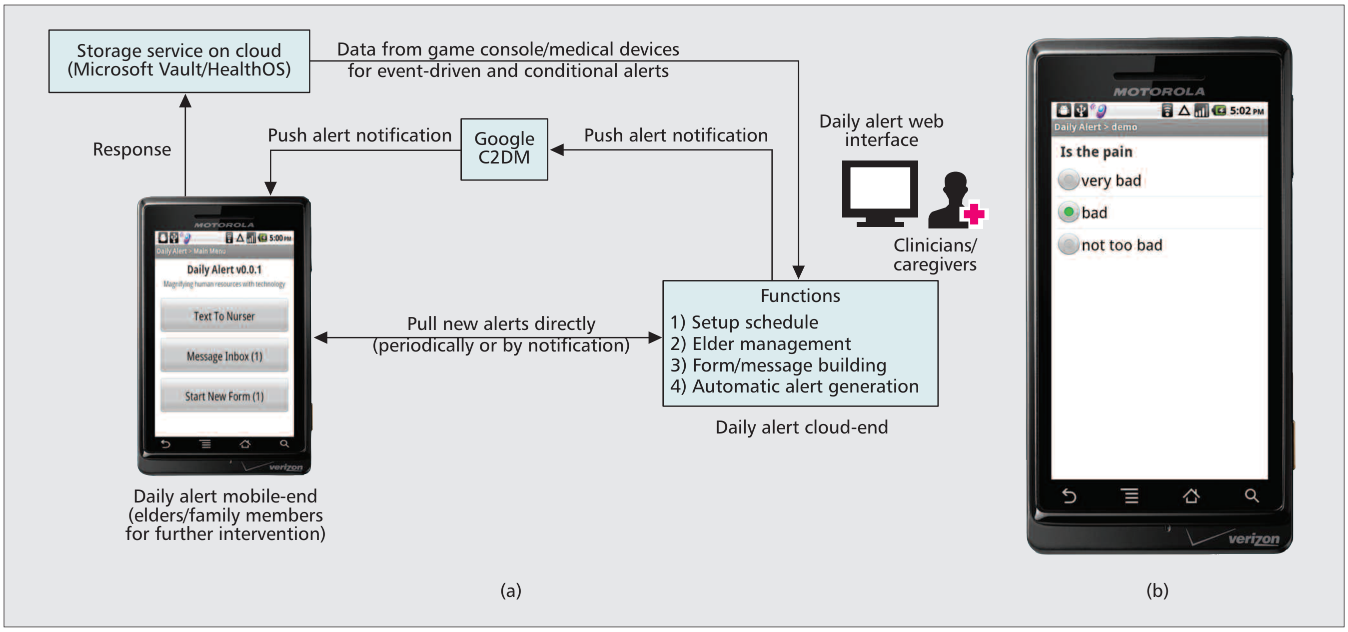


Figure 5. *a) The DailyAlert cloud-end provides caregivers and clinicians with interfaces to define alerts and automatically trigger elders with alerts based on either predefined schedules or analysis of data from external storage; b) the mobile-end benefits elders by providing diverse types of GUI controls (e.g., radio button for click) based on the XML alert form definition. Besides, family members can receive alerts for medication schedules and other management tasks.*

loop approach as a way to improve the wellness of at-home elders using game consoles. A key concept in our closed-loop design is to introduce new components and devices that minimize the cost of care giving, while simultaneously improving the quality of care. Specifically, we incorporate HealthOS and DailyAlert with smartphones into the elder-care intervention loop. These components interact to generate alerts/surveys/questionnaires/notices to the elders' smartphones. Our alerts are user-friendly (e.g., messages and multimedia) and can report status or request actions. HealthOS and DailyAlert help automate the intervention process, provide elders with more communication channels, and include family members in the intervention loop.

We envision that our approach can be later applied to other pervasive healthcare scenarios as well where persistent interventions are required. An example is the care of patients with dementia.

References

[1] L. Martin, V. Freedman, and R. Schoeni, "Trends in Disability and Related Chronic Conditions Among People Ages Fifty to Sixty-Four," *Health Aff.*, vol. 29, no. 4, Apr. 2010, pp. 725–31.

[2] J. Gaugler *et al.*, "Predicting Nursing Home Admission in the US: Ameta-Analysis," *BMC Geriatrics*, vol. 7, no. 13, Jan. 2007, pp. 1471–2318.

[3] E. Lawrence et al., "Interactive Games to Improve Quality of Life for the Elderly," *IEEE ETELEMED '10*.

[4] S. Olson, *The Role of Human Factors in Home Health Care: Workshop Summary*, The Nat'l Academies Press, 2010.

[5] B. J. Fogg, "A Behavior Model for Persuasive Design," *ACM PERSUASIVE '09*.

[6] K. Gerling and M. Masuch, "When Gaming is Not Suitable for Everyone: Playtesting Wii Games with Frail Elderly," *ACM GAXID '11*.

[7] J. Hsu *et al.*, "A 'Wii' Bit of Fun: the Effects of Adding Nintendo Wii Bowling to a Standard Exercise Regimen for Residents of Long-Term Care with Upper Extremity Dysfunction," *Physiother. Theory Prac.*, vol. 27, no. 3, Apr. 2011 pp. 185–93.

[8] W. Dennis *et al.*, "Physical and Psychosocial Effects of Wii Video Game Use among Older Women Physical and Psychosocial Effects of Wii Video Game Use among Older Women Introduction," *Int'l. J. Emerging Tech. and Soc.*, vol. 8, no. 2, 2010, pp. 85–98.

[9] C. Lyles *et al.*, "Qualitative Evaluation of a Mobile Phone and Web-Based Collaborative Care Intervention for Patients with Type 2 Diabetes," *Diabetes Tech. & Ther.*, vol. 13, no. 5, May 2011, pp. 563–69.

[10] D. Estrin and I. Sim, "Open mHealth Architecture: An Engine for Health Care Innovation," *Science*, vol. 330, no. 6005, Nov. 2010, pp. 759–60.

[11] J. H. Lim, A. Zhan, and A. Terzis, "Poster: HealthOS: A Platform for Integrating and Developing Pervasive Healthcare Applications," *ACM Mobisys '10*.

[12] R. Fielding, "REST: Architectural Styles and the Design of Network-based Software Architectures," Univ. CA, Irvine, 2000.

[13] Z. Andong, J. H. Lim, and A. Terzis, "DailyAlert: A Generic Mobile Persuasion Toolkit for Smartphone," *ACM PhoneSense '11*.

[14] C. Hartung *et al.*, "Open Data Kit: Tools to Build Information Services for Developing Regions."

[15] V. Goyal *et al.*, "Attribute-based Encryption for Fine-Grained Access Control of Encrypted Data," ACM CCS '06.

Biographies

JONG HYUN LIM (ljh@cs.jhu.edu) is a Ph.D. student in the Department of Computer Science at the Johns Hopkins University and also a member of the Hopkins InterNetworking Research Group (HiNRG) led by his advisor, Dr. Andreas Terzis. His interests include protocol design, system support, and target localization for healthcare.

ANDONG ZHAN received his Bachelor's and Master's degrees from the Department of Computer Science at Nanjing University. He is a Ph.D. student in the Department of Computer Science at the Johns Hopkins University. He research interests are in applications and systems in wireless sensor networks, smartphone sensing, and mobile healthcare.

JEONGGIL KO received his B.Eng. degree in computer science and engineering from Korea University and his M.S.E. degree in computer science from the Johns Hopkins University, where he is currently working toward a Ph.D. degree. His research interests include wireless medical sensing systems, low-power embedded network system and protocol design, and the deployment of such embedded systems to real environments. He is a recipient of the Abel Wolman Fellowship awarded by the G.W.C. Whiting School of Engineering at the Johns Hopkins University.

SARAH SZANTON received her Ph.D. from the Johns Hopkins University. She is an assistant professor in the Department of Health Systems and Outcomes at the Johns Hopkins University School of Nursing. Her research interests are in development and evaluation of home-based interventions that can help older adults avoid hospitalization and nursing home placement.

LAURA GITLIN is an applied sociologist with over 24 years of funded research on behavioral interventions to improve quality of life of older adults and their family members. She is a professor in the Department of Health Systems and Outcomes at the Johns Hopkins University School of Nursing and director of the newly created Hopkins University School of Nursing Center for Innovative Care in Aging. Her research interests reside in family caregiving, adaptation to functional disability, dementia care, and environmental modification to support daily life.

ANDREAS TERZIS received his Ph.D. degree in computer science from the University of California at Los Angeles. He is an associate professor in the Department of Computer Science at Johns Hopkins University, where he leads HiNRG. His research interests are in the broad area of wireless sensor networks, including protocol design, system support, and data management. He is a recipient of the National Science Foundation (NSF) CAREER award.