iCare: Personal Health Assistant Using Microsoft Azure Cloud

** Krishnaveni S.*
*** Jothi B.*
**** Jeyasutha J.*
***** Sivamohan S.*

Abstract

Nowadays healthcare becomes a critical problem due to lack of availability of expert doctors.In order to overcome this problem we are proposing a personal health Assistant system which will be unified with cloud computing. This paper proposes the iCare Personal Health Assistant from a technical perspective and its benefits for users. iCare personal health assistant monitors patient's health in a very appropriate manner and produces a warning when the patient's vital parameters exceeds the normal value. The histological patient data is transferred to the Azure cloud storage that can be accessed by expert doctors and patients via iOS mobile app. iOS 8 App which can read various health data parameters provided by iOS Health Kit Framework and an Azure Web application which collectively stores all the data to perform simple analytics and report on health conditions for a specific time. The Azure Web application will be developed using ASP.NET and hosted in Microsoft Azure Cloud. Azure Web services and Azure Mobile Services are used to send push notifications. Thus, users get the latest health report along with few suggestive actions to remedy that health condition, if any, and this allows users to adjust their nutrition and activities accordingly.

Keywords: Healthkit API ,Azure Cloud,iOS,Asp.Net, WSDL

I. Introduction

Distributed m-healthcare cloud computing concept has emerged in recent years. We can say that it is a patient centric model as overall control of patient's data is with the patient (Guohua & Malmqvist, 2007). Due to the high cost of building and maintaining data centers, third-party service providers provide healthcare services using third party service providers poses many security and privacy risks. In healthcare social networks, personal health information is always shared among patients located in respective social communities suffering from the same disease for mutual support, and across distributed healthcare providers equipped with their own cloud servers for medical consultation in distributed m-healthcare cloud computing systems [2]. Two intractable problems demanding urgent solutions are patient's personal health information that should be shared and which physician their personal health information should be shared with. In recent years, the distributed m-healthcare (mobile healthcare) has emerged as a paradigm for exchanging health related data of patients. It allows to create, manage and control personal health data, which requires that storage, retrieval, and sharing of medical information is more efficient as in cloud computing [3]. The world healthcare organization defines mobile healthcare system as an area of emerging trends in current health care applications that provides health information and services over mobile technologies such as mobile phones and Personal Digital Assistants (PDAs). Personal health information is always shared among patients suffering from the same disease, between patients and physicians as equivalent counterparts or even across distributed healthcare providers for medical consultation. This kind of personal health information sharing allows each collaborating healthcare provider to process it locally with higher efficiency and scalability, greatly enhancing treatment

**Assistant Professor,* Software Engineering, SRM University, Chennai - 603 203. E-mail : vanimithila@gmail.com
***Assistant Professor,* Software Engineering, SRM University, Chennai - 603 203. E-mail : jothi.sundari@gmail.com
****Assistant Professor,* Software Engineering, SRM University, Chennai - 603 203. E-mail : jjeyasudha@gmail.com
*****Assistant Professor,* Information Technology, SRM University, Chennai - 603 203. E-mail : sivamohan7@gmail.com

quality, significantly alleviating complexity at the patient side, and therefore, becoming the preliminary component of a distributed m-healthcare system [4].

II. Related Work

iCare is a personal health-care web application and mobile-based system which can read various health data parameters provided by iOS Health Kit Framework which is collected by using various health devices and an Azure Web application which collectively stores all the data to perform simple analytics and prediction of health conditions for a foreseeable time. The objectives are prediction of health conditions well in advance with high level of accuracy, no scope limitations, easy combat actions which go on with daily life routines. To understand the user's physical limits and plan actions accordingly, medical history based analysis is to be carried out by the system. Everything will be approved and certified by a nutritionist to ensure no harm is caused by usage of the system. iCare web application is developed using ASP.NET and hosted in Microsoft Azure cloud. A comparing engine will intelligently analyze the standard recommended data store and map it with the current health readings of the user to predict deviations in health conditions. Thus, the user can follow suggested actions to remedy the predicted health condition. It suits all age groups irrespective of their physical characteristics. The proposed iCare healthcare system consists of two parts. First, Azure Mobile Services are used for connectivity to the iCare iOS App using predefined APIs. Azure push notification service is used for sending frequent health reports to the iOS App. The iOS app is developed using two languages i.e. Swift and Objective-C. The Health kit API is used to check the health data from Health kit store in iOS. The health data is categorized into five main health categories including nutrition, exercise, body vitals, body fitness, and sleep. These categories consist of many available health parameters which can be utilized to keep track of health conditions with a high level of accuracy. After the data is imported, a copy of it will be sent to the Web app in Microsoft Azure Cloud for further processing and storage. The storage of data does not violate health laws applicable to cloud storage as only the calculated inference data and the individual health score will be stored. The raw health data will never stored for any user. This type of processing can lead to dynamic health scores which can change rapidly over time, in days or in hours or even in minutes, as required by the user. The health data can also be used to suggest simple actions to bring the level of health impact to normal, if found to be very high or very low. The report is then generated with health parameters and its levels of impact on the overall health, which is then push notified to the iOS app. Thus, the user can get a timely notification of a snapshot of current health condition and plan his/her nutrition and activities for the day accordingly.

III. Existing Systems

The existing health applications have a drawback of reading and storing health data for limited time period only. So, there is no prediction based implementation had been done in existing system [5]. Hence, computational and comparative accuracy is very low. People are not able to understand their health conditions from a snapshot of data, and because of the above mentioned reason patient should consult a doctor to diagnose and analyze the causes of illness [6]. Also, there are no nutritionist approved standard charts to view and analyze live health conditions automatically [7]. Centralized health data stores are available which provide at a glance look at the present health condition but these are limited by scope.

IV. Literature Survey

A literature study was conducted and it is summarized in Table I.

V. Proposed System

The proposed system focuses on collection of patient's vital health parameters to generate alert warning to healthcare takers so that immediate remedy action can be taken in case of emergencies. The data is then stored in Cloud so that data can be accessed through internet from anywhere anytime. Modern technology is being moved to cloud based platform as it is best suited for long-term data storage. The objectives of iCare are prediction of health conditions well in advance with high level of accuracy, no scope limitations, easy combat actions which go on with daily life routines. iCare will use web services description languages calls for querying and storing data between iOS app and Azure web application. Functional requirements are 128-bit Azure encryption and authorization, password recovery mechanisms, A

TABLE I. LITERATURE REVIEW

S.No	Title / Publication	Data Sets / Techniques Adopted	Functions	Conclusions
1	A Mobile Health Monitoring System for the Elderly - IEEE, National Science Forum, China - 2013	Body Sensor Networks (BSNs) with short range communication with PDA.	Real-Time data collection. Automatic alerts to family and relatives.	Manual health readings must be taken and entered to system.
2	A Survey on Ambient Intelligence in HealthCare Proceedings of the IEEE - 2012	Body Area Networks (BANs) with hierarchal tiers of devices and sensors. Ambient Intelligence Systems (AmI) with behavioural pattern recognition and continuous monitoring.	Accurate readings of health parameters (crucial only). Critical dependencies of health parameters is uncovered.	No support for Smart-Wearable devices / Requires purchase of body sensors such as ECG etc.
3	A Cloud Computing Solution for sharing healthcare information - Proceedings of the IEEE - 2012	Google APP Engine(GAE) based desktop system connecting health centres and hospitals. Large scale binary file sharing is used to share larger media files like patient X-Ray images.	A comprehensive solution to share critical health data between hospitals and health centers, with allowable read access to patients.	Although read access is provided to patients, real-time data is not used to monitor past health situations,.
4	Mobile Healthcare Service System Using RFID - Proceedings of the IEEE - 2004	RFIDs hold a patient's disease information and smart generators scan the RFIDs. Infection control system in both hospital and local communities.	Very quick analysis and scanning of medical history. Prevention of diseases which are spreading too fast in a community can be controlled by analyzing large amounts of data.	Even though high-speed Wireless networks are used, data transfer speed cannot exceed 10MB/s.

Microsoft SQL for Azure database capable of handling 1000 queries per minute, access to Azure database by the web application 24/7, simple data comparison techniques, access to iOS app through WSDL calls, prediction reports and analysis results pushed to the device. Non-functional requirements for performance are: minimum infrastructure requirements on Cloud CPU 0.5 GHZ or higher, memory of 0.5 GB or higher, 150MB of free storage space, 500,000 API calls every 3 months or higher.

Few advantages that iCare has over other systems include:

❖ This will be the first cloud based system with a snapshot of all health parameters, not limited by scope.

❖ Fits in your pocket, as an iOS app running in the background.

❖ The iOS app will recover all the data stored in Health Kit momentarily and not just a few of them.

❖ Since iOS and Health Kit is used, encryption will be strong and user information always stays private.

❖ The power of Azure cloud means that the system can scale depending on the load and never crash.

❖ Data accuracy will be high (subject to smart device limitations) because every action predicted by the system is verified with a nutritionist and it consists of only simple everyday activity and nutrition remedies.

❖ It suits all age groups irrespective of their physical differences, as all their physical data will be extracted from the users and then processed.

❖ It allows adjusting the user's body limits to align actions accordingly.

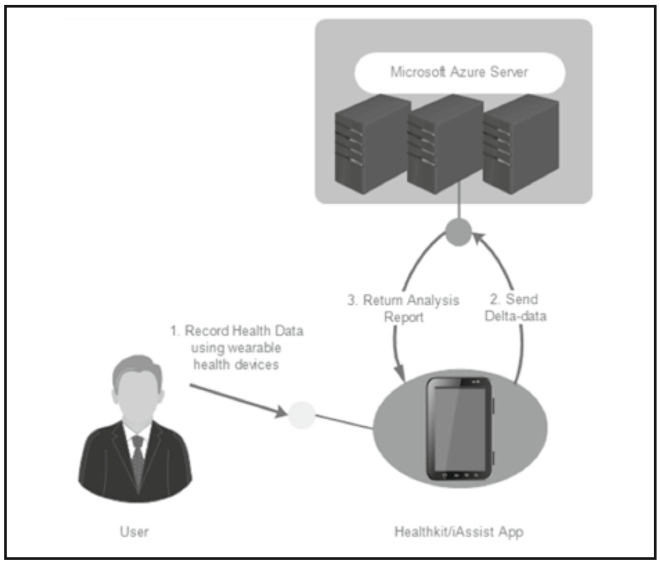

Figure 1. iCare Personal Health Assistant System Architecture

VI. System Architecture

Detailed Design

Health kit Recovery Component

HealthkitRecovery of iCare iOS App allows usage of Apple's iOS Health kit APIs to automatically input health data in terms of various parameters stored in the Health kit app. User privacy to be high as only health parameters which are allowed by the user to be read can be accessed. However, crucial data required for the app to work will be flagged as mandatory. The raw health data to be stored temporarily in the main memory for further processing .
Input: Raw health data in parameters (with units) which are read by Health kit API
Output: Categorized health data in main memory, kept by iOS app

AzureCompute Component

AzureCompute will read the health data stored in main memory and compare it with the standard health norms (which are entered by the user during initial startup of app). This comparison allows the calculation of delta-data (the difference in readings either positive or negative) with respect to the actual health readings. This delta-data is uploaded to the remote Azure Server using Azure Mobile-Services API. Then is stored in the Microsoft SQL Server for Azure in the remote Azure server. This component also provides login facility to the user via the app. Azure Authorization and Authentication mechanisms with 256-bit encryption using AES protocol (standard for any Azure Server) implemented here. After successful authorization, the delta-data is uploaded to the server. Logging in using Microsoft accounts and recovering passwords via e-mail also done here.
Input: Health data in main memory, kept by iOS app
Output: Delta-data stored in SQL database in remote Azure Server

AzureAnalysis Component

AzureAnalysis of iCare ASP Web Application take as input,the delta-data stored in the SQL database along with standard normal health data by average (given by a physician) and analyses these data by comparing the health parameters with one another. Included are the mechanisms to analyze all health parameters instead of relying only on those necessary parameters to calculate health percentage (as a prediction report) for any kind of disease or regular checkups. The prediction report is meant to determine the appropriate suggestive actions to be sent to the user on a timely basis. These suggestive actions (pre-entered into the app) are read appropriately by the Web app and passed on to the AzureResult component.
Input: Delta-data along with standard health average (given by a physician)
Output: Suggestive action

AzureResult Components

AzureResult takes as input the suggestive actions passed by the Azure Analysis component and uses the WSDL based Azure mobile services API to push notify the iCare iOS app with the suggestive action. The suggestive actions are queued and sent through the network by the same API.
Input: Suggestive actions provided by AzureAnalysis
Output: Push notification with the suggestive action

AzureRecovery Component

AzureRecovery is designed as a manual way of getting a suggestive action based on a push notification. This also uses the Azure mobile services API to poll the server for any suggestive actions that are waiting to be sent to the iOS app, but not yet sent due to network outage. The user has to click a button to pull the suggestive action from the server to the app.
Input: suggestive actions queued to be sent by AzureResult
Output: Manually pulled suggestive action displayed in iOS app

VII. Conclusion

This paper proposes a iCare Personal Health Assistant Systems based on mobile cloud computing environment. This provides a high level of integration, interoperability, and sharing for healthcare providers, patients and practitioners. The cloud permits fast Internet access and sharing by authenticated users. The ultimate goal of the proposed system is to introduce a new generation of HealthCare systems that provide healthcare services of high quality and low cost to the patients using a combination cloud computing and mobile computing technologies. The system to be developed consists usage

SYSTEM ANALYSIS:

Requirement ID	Description	Priority
RR1	The System should allow the user to login using 128-bit Azure standard login	High
RR1a	Logging in using Microsoft ID is possible	Low
RR1b	Password Recovery is through E-Mail	Medium
RR2	With the click of a button, iOS app should recover all the data from Healthkit	High
RR2a	User privacy over types of health data is to be strictly given	High
RR2b	Mandatory types of data based on health problem is to be strictly imposed to the user	Medium
RR2c	Storing all the health data locally is prohibited	High
RR3	Uploading data to Azure database is done only on the delta-data calculated	High
RR3a	Delta-data is calculated based on the comparison between norms and the actual health readings over time	High
RR3b	Delta-data upload is through Azure Mobile Services only	Medium
RR3c	The Azure database should be capable of handling 1000 queries per minute or lower as dictated by network speed	Low
RR4	Analysis and prediction logic with suggestive actions is to be implemented in the Azure website	High
RR4a	Comparison of delta-data with historical data of the individual and the standard norms is to be used in prediction	Medium
RR4b	Prediction report generated with a time limit of expiry and accuracy limits	High
RR4c	Simple suggestive actions are chosen based on prediction report	High
RR4d	Data discrepancy problems to be avoided (harmlessness)	Medium
RR5	Medical history is mandatory to be entered by the user	High

of publicly available frameworks and APIs (Application Programming Interfaces), in an intuitive way. The harmless nature of combat steps adds to its uniqueness. It will work according to the user's body limits and pace and continued usage will only lead to higher accuracy. Hence, it will be a true “Health in your Pocket” implementation with many extended applications. Some of the planned enhancements for this system include the prediction of impending health failures/diseases and combating with simple everyday steps if health situation is perfectly normal, suggestion of global WHO recommended healthy living actions as per the user's pace. Understanding the user's response times and aligning actions and malfunctions in health conditions detected to be provided well in advance along with recommended healthy-living simple actions to combat those malfunctions.

References

[1] G. Bai and G. Malmqvist, “Guide to good practice of ehealth,” Research Report, IANIS+GGP ehealth Group, 2007.

[2] T. Martin, E. Jovanov, and D. Raskovic, “Issues in wearable computing for medical monitoring applications: A case study of a wearable ECG monitoring device,” Int. Symp. Wearable Computers ISWC 2000, Atlanta, GA, Oct. 2000.

[3] A. Pantelopoulos and N. G. Bourbakis, “Prognosis - a wearable health- monitoring system for people at risk: methodology and modeling,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 3, 2010.

[4] A. Milenkovic, C. Otto, and E. Jovanov, “Wireless sensor network for personal health monitoring: Issues and an implementation,” *Computer Communications*, vol. 29.

[5] M. J. Gregoski, M. Mueller, A. Vertegel, A. Shaporev, B. B. Jackson, R. M. Frenzel, S. M. Sprehn, and F. A. Treiber, “Development and validation of a smartphone heart rate acquisition application for health promotion and wellness telehealth applications,” *International Journal of Telemedicine and Applications*, vol. 2012.

[6] American Heart Association. Available: http://www.americanheart.org, 2007.

[7] D. Lloyd-Jones et al., “Heart disease and stroke statistics—2009 update: a report from the American heart association statistics committee and stroke statistics subcommittee,” Circulation 2009; 119: 41. doi: 10.1161/CIRCULATIONAHA.108.191261

[8] Y. M. Huang, M. Y. Hsieh, H. C. Chao, Hung, S. H., and Park, J. H., "Pervasive, secure access to a hierarchical sensor-based healthcare monitoring architecture in wireless heterogeneous networks," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 4. doi: 10.1109/JSAC.2009.090505