An IoT-Based Framework for Health Monitoring Systems: A Case Study Approach

N. Sudhakar Yadav, Depart of CSE, Jawaharlal Nehru Technological University Anantapur, Andhra Pradesh, India
K. G. Srinivasa, CBP Government Engineering College, New Delhi, India
B. Eswara Reddy, JNTUA College of Engineering, Kalikiri, India

ABSTRACT

A software framework is a reusable design that requires various software components to function almost out of the box. To specify a framework, the creator must specify the different components that form the framework and how to instantiate them. Also, the communication interfaces between these various components must be defined. In this article, the authors propose such a framework based on the internet of things (IoT) for developing applications for handling emergencies of some kind. This article demonstrates the usage of the framework by explaining various applications such as tracking the status of autistic students, analytics on medical records to detect and mitigate chronic heart diseases in the Indian demographic, prediction of Parkinson’s disease, determining the type of disease that corresponds to the dermatology field, and health monitoring and management using internet of things (IoT) sensing.

KEYWORDS

Health Monitoring, Internet of Things (IoT), Medical Records, Prediction of Disease

INTRODUCTION

The Internet of Things (IoT), also called Internet of Everything is the network of physical objects or “things” embedded with electronics, software, sensors, and connectivity to enable objects to exchange data. The Internet of Things allows objects to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integration between the physical world and computer-based systems, and resulting in improved efficiency, accuracy and economic benefit (Atzori, Iera, & Morabito, 2010).

IoT devices can be used to enable remote health monitoring and emergency notification systems. These health monitoring devices can range from blood pressure and heart rate monitors to advanced devices capable of monitoring specific patients ensuring that proper treatment is being administered. This paper provides an IoT based framework to handle medical emergencies and takes up case studies to analyze the application of this framework.

The architecture (Refer Figure 1) of the framework is divided into four modules:

DOI: 10.4018/IJFC.2019010102

Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
Data Collection

An important part of creating a framework for IoT is the collection the raw data from the environment (by means of various sensors) and sending it over to the cloud for storage and analysis. Our framework requires an engineer to just plug the sensors required to his microcontroller and write sensor specific code to access the data. This paper provides the architecture for handling of various sensors and services to send over data to the cloud. Every data collection unit is comprised of a microcontroller and multiple sensors attached to it. Sensors A sensor is a transducer whose purpose is to sense (that is, to detect) some characteristic of its environments. It detects events or changes in quantities in it’s environment and provides a corresponding output, that needs to be interpreted by the user of the sensor. For example, a GPS (Global Positioning System) sensor provides the location of where it is placed, in terms of latitude and longitude; it is up to the user of the sensor to access the output and interpret it the way he likes.

Our framework provides a base ISensor class that is the fundamental structural unit of the entire data collection module. This class has two functions, namely readData() and getData().

- readData(): Reads the data from the sensor, and stores it in a serialized form.
- getData(): Getter method for the serialized sensor data.

Every sensor has its own concrete class, that helps to access the data, which needs to inherit the base ISensor class and define the above mentioned functions. We take care of all the instantiations and provide each sensor's object in the main controller program.

Microcontroller

A microcontroller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. The specifications of the processing power and memory vary from one microcontroller to another, but they are generally sufficient to run our light framework. Our entire data collection module is run on the microcontroller. All the sensors need to be connected to the microcontroller. The main controller program runs a forever loop and accesses data from each of the connected sensors or as the engineer wishes to collect the data, serializes it and sends it over to the cloud, at regular intervals of time.

REST Services

Representational State Transfer (REST) is a software architecture style consisting of guidelines and best practices for creating scalable web services. REST is a coordinated set of constraints applied to the design of components in a distributed hypermedia system that can lead to a more performant and maintainable architecture. RESTful systems typically communicate over the HTTP with the same HTTP verbs (GET, POST, PUT etc.)

The entire communication between various components in our framework is built on top of these REST services. We provide a ready to use, simple REST API on the cloud, where the engineer only needs to code how various end points are supposed to behave. We have also integrated Android PUSH notification support into the services, which can be easily triggered as and when required.
A Study on the Performance and Scalability of Apache Flink Over Hadoop MapReduce
www.igi-global.com/article/a-study-on-the-performance-and-scalability-of-apache-flink-over-hadoop-mapreduce/219361?camid=4v1a