Journal of The Institute of Electronics and Information Engineers Vol.53, NO.12, December 2016 https://doi.org/10.5573/ieie.2016.53.12.152
ISSN 2287-5026(Print) / ISSN 2288-159X(Online)

논문 2016-53-12-19

MongoDB를 활용한 풀 스택 플랫폼 설계

(Full Stack Platform Design with MongoDB)

홍 선 학*, 조 경 순*

(Hong Sun Hag© and Cho Kyung Soon)

요 약

본 논문에서는 오픈소스 플랫폼 라즈베리파이 3 모델을 기반으로 몽고DB 데이터베이스를 활용하여 풀 스택 플랫폼을 구현하였다. 가속도 센서를 사용하여 무선 통신으로 데이터를 로깅하는 도구로써 이벤트 구동 방식을 사용하였으며, 리눅스 라즈비안 Jessie 버전으로 초당 28 프레임으로 USB 카메라(MS LifeCam 시네마) 이미지를 획득하며, 안드로이드 모바일 기기와 인터페이스를 구축하기 위하여 블루투스 통신 기술을 확장하였다. 따라서 본 논문에서는 가속도 센서 동작을 검출하여 이벤트 트리거링을 감지하는 풀 스택 플랫폼 기능을 구현하고, IoT 환경에서 온도와 습도 센서 데이터를 수집한다. 특히 몽고 DB가 MEAN 스택과 가장 좋은 데이터 연결성을 갖고 있기 때문에 풀 스택 플랫폼 성능을 개발 향상시키는데 MEAN 스택을 사용하였다. 향후 IoT 클라우드 환경에서 풀 스택 성능을 향상시키고, 몽고 DB를 활용하여 보다 쉽게 웹 설계 성능을 향상시키도록 기술을 개발하겠다.

Abstract

In this paper, we implemented the full stack platform design with MongoDB database of open source platform Raspberry PI 3 model. We experimented the triggering of event driven with acceleration sensor data logging with wireless communication. we captured the image of USB Camera(MS LifeCam cinema) with 28 frames per second under the Linux version of Raspbian Jessie and extended the functionality of wireless communication function with Bluetooth technology for the purpose of making Android Mobile devices interface. And therefore we implemented the functions of the full stack platform for recognizing the event triggering characteristics of detecting the acceleration sensor action and gathering the temperature and humidity sensor data under IoT environment. Especially we used MEAN Stack for developing the performance of full stack platform because the MEAN Stack is more akin to working with MongoDB than what we know of as a database. Afterwards, we would enhance the performance of full stack platform for IoT clouding functionalities and more feasible web design with MongoDB.

Keywords : Full stack platform, Event driven recognition, MongoDB 3.2, MEAN Stack, RPi 3

Ⅰ. 서 론

In the last 20 years, the Internet has challenged relational databases in ways nobody could have foreseen. Having used MySQL at large and growing Internet jobs during these times, we've seen this happen increasingly. First we had a single server with a small data set. Then we found ourselves setting up replication so we can scale out reads and deal with potential failures. And therefore before too long, we have added a caching layer, tuned all the queries, and thrown even more hardware at the problem. Eventually In this paper, we have drawn to MongoDB with balancing between features and complexity toward making previously difficult tasks in SQL database families far easier.

We used MongoDB for the purpose of tools as a powerful, flexible, and scalable general-purpose database. It combines the ability to scale out with

* 정회원, 서일대학교 컴퓨터응용과 교수
© Corresponding Author (E-mail: hongsh@seoil.ac.kr)
※ 본 연구는 2016학년도 서일대학 교내학술연구비지원으로 수행되었음.
Received ; July 25, 2016 Revised ; August 12, 2016
Accepted ; November 30, 2016

features as a secondary indexes, range queries, sorting, aggregation, and geospatial indexes. In this paper, we mainly try to take care of the connection points between existing popular frameworks and solve common integration efforts for capturing image at the moment of triggering acceleration sensor action under MEAN stack.

MEAN stack means the abbreviation of MongoDB, Express, AngularJS and Node.js. MongoDB is the leading NoSQL database with empowering performance to be more agile and scalable. Express is a minimal and flexible Node.js application framework, providing a robust set of features for building single and multi-page, and hybrid web applications. AngularJS lets us extend HTML vocabulary for your application. The resulting environment is extraordinarily expressive, readable, and quick to develop. And therefore Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications.

In this paper, we used the Raspberry Pi 3 Model , an USB camera(MS LifeCam) and acceleration sensors(GY-521), with an 16 GB SD card, and the Raspbian Jessie OS because it has the advantage of cost and performance respects against other platforms. Figure 1 showed the overview of MEAN stack architecture system which had an server including Node.js, Express, and mongoose, an client named AngularJs and MongoDB for implementing the real time image capturing under the Linux OS Raspbian Jessie.

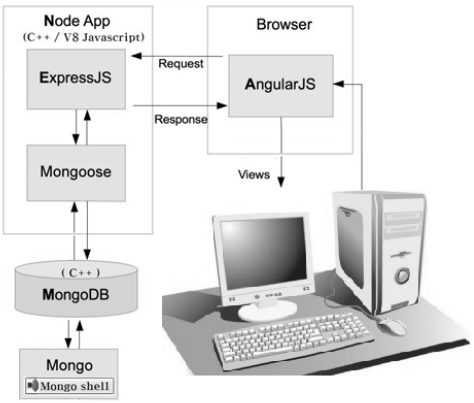

Fig. 1. Overview of MEAN stack system.

Especially, we would make MEAN stack helpful for developing the combination of between USB camera and accelerometer sensors and realtime processing through bluetooth communication technology under the IoT environment[1~3].

Ⅱ. Basic Theory

2.1 Overall System Structure

In this paper, we used the embedded platform which has the UART, TCP/IP and Web streaming services with MEAN stack. Figure 2 showed the flow of Initialization system which has the functions of logging, accelerometer execution, serial communication with UART, FTP server and TCP operation as soon as the power on.

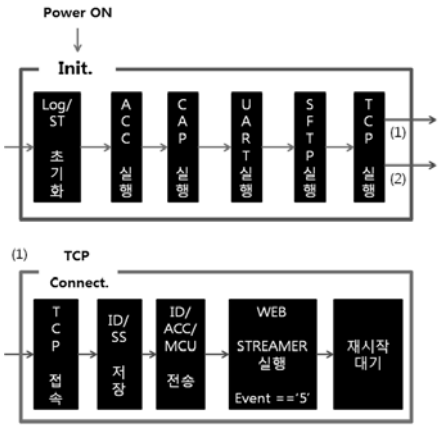

Fig. 2. Flow of Initialization system.

We used Raspberry Pi 3 model. It runs the same software, and still has 1GB RAM with Camera interface. We used the most common Raspberry Pi distribution open source operating system "Raspbian Jessie" which can run the full range of ARM GNU/Linux distributions. USB camera module could be used to take high-definition video, as well as stills photographs.

Figure 3 showed the flow of Algorithm procedure which event calculation, image frame accumulating, UART status transmission and TCP/IP communication. When an event happened, platform loaded the

accelerometer data after 3 seconds, at the same time camera image and speed saved and released the camera operation, and therefore MCU read the event data and saved the speed. After event 4 happened, TCP/IP communication waited for sending the event information to the server[4~6].

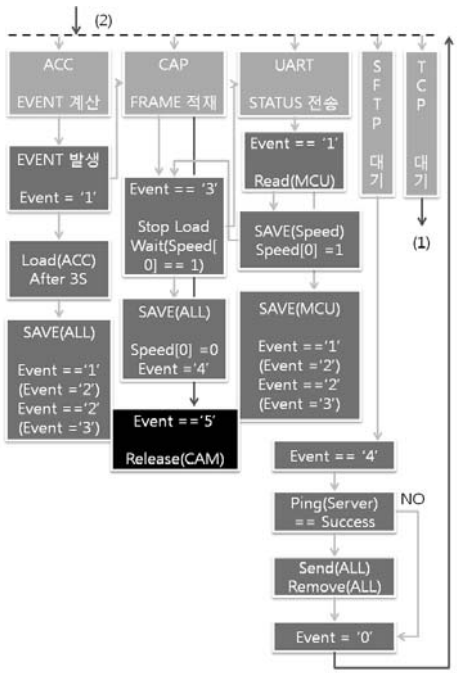

Fig. 3. Procedure of Overall algorithm.

2.2 Raspberry Pi Features

The Raspberry Pi 3 is the 3rd generation Raspberry Pi series have a 1.2GHz 64-bit quad-core ARMv8 CPU, 802.11n Wireless LAN, Bluetooth 4.1(Bluetooth Low Energy). The Raspberry Pi 3 has an identical form factor to the previous Pi 2 (and Pi 1 Model B+) and has complete compatibility with Raspberry Pi 1 and 2.

It has also a 1GB RAM, 4 USB ports, 40 GPIO pins, Full HDMI port, Ethernet port, Camera interface (CSI), Display interface (DSI), Micro SD card slot, VideoCore IV 3D graphics core. We used Digital GPIO 6 pins for connecting the accelerometers for the purpose of detecting the shaking of platform at the same time camera captured the image during 3 seconds.

2.3 MongoDB Features

In this paper, we used MongoDB which is popular for a general-purpose database, so aside from creating, reading, updating, and deleting data, it has an ever-growing list of unique features : It supports generic secondary indexes, allowing a variety of fast queries, and provides unique, compound, geospatial, and full-text indexing capabilities as well. And also it supports time-to-live collections for data that should expire at a certain time, and supports fixed-size collections which are useful for holding recent data such as logs, and finally supports an easy-to-use protocol for storing large files and metadata.

Data in MongoDB has a flexible schema. Unlike SQL databases, where you must determine and declare a table's schema before inserting data, MongoDB's collections do not enforce document structure. This flexibility facilitates the mapping of documents to an entity or an object. Each document can match the data fields of the represented entity, even if the data has substantial variation. In practice, however, the documents in a collection share a similar structure.

The key challenge in data modeling is balancing the needs of the application, the performance characteristics of the database engine, and the data retrieval patterns. When designing data models, always consider the application usage of the data (i.e. queries, updates, and processing of the data) as well as the inherent structure of the data itself. Figure 4 represented the comparisons between performance and extensibility of databases. MongoDB has more

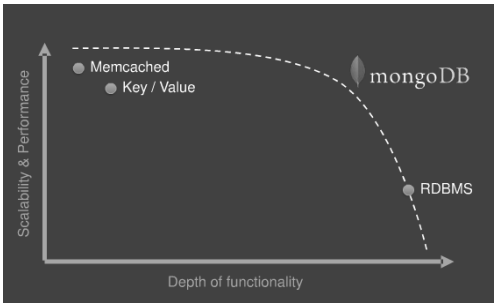

Fig. 4. Comparisons of performance.

excellent functionality and performance than other SQL databases.

Ⅲ. Features of Platform

In this paper, we used the Pi camera board plugs directly into the CSI connector on the Raspberry Pi 3. It's able to deliver a crystal clear 5 mega pixel resolution image. The module attaches to Raspberry Pi to the dedicated 15 pin MIPI Camera Serial Interface(CSI), which was designed especially for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to main-processor.

3.1 Physical Platform

In this paper we captured the image of PRi camera at the rate of 25~29 frames per second, loaded the image on the RAM and save the image files. We defined the resolution of Frame_width, height in the processCommandLine function and used 1280*906 resolution on the experiment. Figure 5 showed the acquired image of Rapsberry Pi 3 camera for 26.9558 frames per second[7~9].

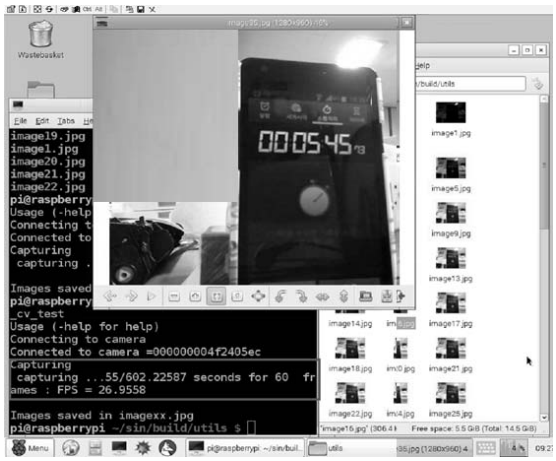

Fig. 5. Acquired image of RPi camera.

3.2 Software Platform

We used the MEAN stack which is an opinionated full stack javascript framework which simplifies and accelerates web application development. It is designed to give us a quick and organized way to start developing MEAN based web apps with useful modules like Mongoose and Passport pre-bundled and configured. We mainly try to take care of the connection points between existing popular frameworks and solve common integration problems[10~12].

Figure 6 represented the overall plan and interactions between the various system in the MEAN stack. It will consist of four main parts: the log-in or sign-up form, a MongoDB database, and a server we'll create with JavaScript and run on Node.js.

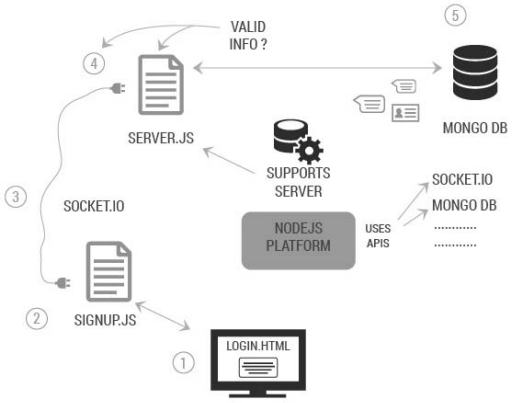

Fig. 6. Features of MEAN stack.

The login screen – a basic HTML form – will take the information a user enters and pass it to the server. The server will perform all the validation on the data, and if there are no errors, it will submit the data as a JSON object to the database. Otherwise, it will pass an error message back to the client to display as a JavaScript alert to the user.

MongoDB is an open-source, cross-platform database that we'll use to store our profile information and messages. Instead of a traditional database that stores information as tables, MongoDB give us the ability to upload data as JSON objects in documents using BSON.

Node.js is an open-source runtime environment developed by Ryan Dahl, first published in 2009. An important distinction is that Node isn't a server; it's a

platform that allows you to create a JavaScript-based server. It's open source, so we can take advantage of huge community of developers that contribute to it's growing library. Both server-side and client-side apps are written in JavaScript. A huge advantage here is that developers can get started quickly, on a powerful platform, using a programming language that they're already familiar with. The NPM, or Node Package Manager, let's quickly upload or download their libraries to or from the wide community of open-source developers. Additionally, one of the most distinguishing benefits of Node.js is its incredible speed, due to its event-driven architecture and non-blocking I/O model. This model allows thousands and thousands of simultaneous connections. Figure 7 shows the asynchronous programming method. In asynchronous programming, Function I/O procedures run on separate threads in parallel with main main processing thread.

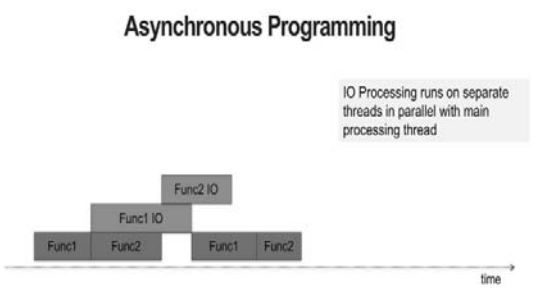

Fig. 7. Asynchronous Programming model.

3.3 MongoDB Platform

In this paper, we used an open-source database MongoDB that provides index and geospatial information. MongoDB obviates the need for an Object Relational Mapping(ORM) to facilitate development. A record in MongoDB is a document, which is a data structure composed of field and value pairs. Geospatial indexes allow us to store geospatial coordinates in the data and do searches based on the proximity of one point to another. Figure 8 shows the location of SEOIL University in the geospatial index in MongoDB[13~14].

```
db.places.find({location:{$near:{$geometry:{type:"Point",
coordinates:[37.5866211, 127.095555]}}}})
```

MongoDB used the coordinates which are represented as longitude and latitude. Logitude measures from Greenwich meridian in London(0 degrees) locations east and for locations west we specify as negative. Latitude measures from equator north and south. Coordinates in MongoDB are stored on Longitude/Latitude order.

Fig. 8. Geospatial Location in MongoDB.

Ⅳ. Platform Feasibility Design

In this section, we described the characteristics of full stack MongoDB platform. The basic part of platform is that opensource software MEAN Stack with MongoDB and accelerometer for gathering events signal data and for displaying real time information via the bluetooth communication module[15~17].

4.1 FTP Server Architecture

The proposed platform took a modular type for accumulating the individual modules in order to extend the abilities. Figure 9 showed the flow of SFTP file transferring with system functions.

We used system function for sending SFTP files, confirmed the internet connection by finding the ping results, and would designate the IP address with reading number "3" in the Event.txt file. And

therefore FTP server received the files and removed the file in the folder[18~19].

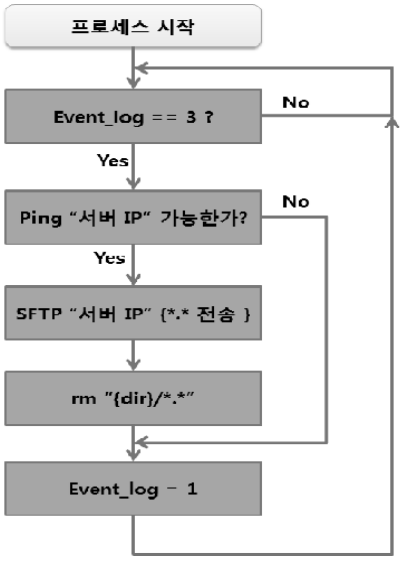

Fig. 9. FTP server architecture.

4.2 TCP/IP Design Layout

In this paper, We made the TCP/IP layout like Figure 10. There were Rapsberry Pi as a TCP server and client as a port 127.0.0.2. We made a client with designated server address by executing the client cpp file. When the server received the packet from client, and server returned the string of specific file as a packet to the client. Figure 10 showed the TCP/IP design layout.

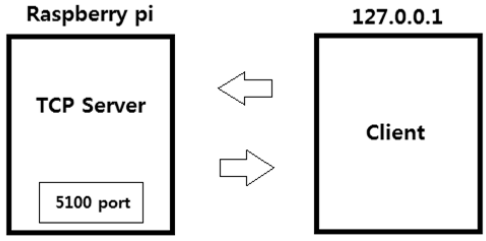

Fig. 10. TCP/IP Design layout.

4.3 Experiments for Vision Platform

We experimented the event detection platform with Bluetooth communication through MongoDB of MEAN Stack. The number of event and action time sensor attached on platform measured the data every specified period with MongoDB database for logging in time base[20~21].

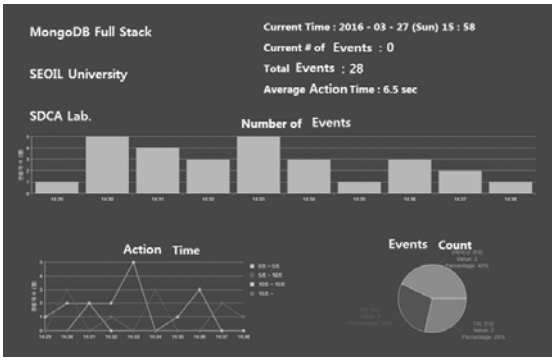

Fig. 11. Event Recognition with Web server.

Figure 11 represented the event recognition of Raspberry Pi 3 camera captured image with MongoDB.

V. Conclusions

In this paper, we implemented the event detection with USB camera in Rapsberry Pi with BLE. Especially we directly designed the web browser with MEAN stack. Afterwards we would enhance the performance of full stack platform for clouding functionalities.

REFERENCES

[1] Park Jae Ho, Modern App Development Basic with MEAN Stack, Acorn, 02.2015.
[2] Son Byung Dae, Node Web Development,, Acorn. 11, 2011.
[3] https://nodejs. org/documentation/Tutorials
[4] Hong. Sun Hag and Choi Young Woo, Opensource MongoDB with studying Node.js & fluentd, SungAnDang, 03, 2016
[5] Leo Breiman, "Random Forest, Machine Learning," vol. 45, 2001, pp5~32.
[6] Paul Viola and Michael J. Jones, "Robust real-time face detection," International Journal of Computer Vision, Vol 57, No2, 2004, pp137~154.
[7] Hong Sun Hag, "Monbile Embedded USN Platform Design" Korea Institute of Communications and Information Sciences, 37

vol, 4th, 08, 2012.
[8] Oh Il Suk, Computer Vision - From Basic to Recent Mobile Application, Hanbit Media, 07, 2014
[9] Hong Sun Hag, Cho Kyung Soon, “Computer Vision Platform Design with MEAN Stack,” Korea Society of Digital Industry & Information Magement, 11 vol, 3th, 2015, 09, pp. 79-87.
[10] Hong Sun Hag,, “Mobile Arduino Embedded Platform Design,” Korea Society of Digital Industry & Information Magement, 9 vol, 4th, 12, 2013, pp. 33-41
[11] Kim Young Ro, Cho Young Tae, Ryu Seung Gi “Pothole Detection using Intensity and Motion Information”, Institute of Electronics and Information Engineers, 52(11), 2015.11 pp137-146.
[12] Seo Choo Won, ”The Image Position Measurement for the Selected Object out of the Center using the 2 Points Polar Coordinate Transform“ Institute of Electronics and Information Engineers,, 52(11), 2015.11 pp147-155.
[13] Mongo DB Web, http://docs. mongodb. org/manual/, 08,2015.
[14] Michael McTear, Callejas Zoraida, Voice Application Development for Android, Acorn, 08.2014.
[15] Hong Sun Hag, Arduino following with Mobile texhniques, SungAnDang, 04, 2013
[16] Wolfgang Mauerer, Professional Linux Kernel architecture, Wiely Pub, 2007.
[17] Downey, Allen B, Python for software Design, Cambridge, 03, 2009
[18] Simon Monk, RapsberryPi Cookbook, HanBit Media, 01, 2015
[19] Robert Lagani Re,, OpenCV 2 Computer Vision Application Programming Cookbook, Acorn,04, 2012
[20] Computer Bookshops, Opencv Computer Vision Application Programming Cookbook (2nd Edition), Packet Pub.03, 2013
[21] Drew Conway, John Myels White, ″Machine Learning for Hackers″, Insight, 01, 2014.

저 자 소 개

홍 선 학(정회원)
1992년～현재 서일대학
컴퓨터응용과 교수
1994년 광운대학교 대학원
박사 졸업

<주관심분야 : 임베디드 리눅스, IoT 클라우딩

조 경 순(정회원)
1998년～현재 서일대학
컴퓨터응용과 교수
2008년 광운대 대학원 박사 졸업

<주관심분야 : 제어 응용, 전기전자재료>