A Design of Web-based Services Using RESTful API for Vertical Farm

Saraswathi Sivamani, Kyunghun Kwak, Yongyun Cho

Department of Information and Communication Engineering, Sunchon National University, Sunchon, Jeonnam 540-742, South Korea
{saraswathi, supersdar, yycho}@sunchon.ac.kr

Abstract: With the development of human innovation, quality and quantity of food production have been increased in agriculture, thanks to the vertical farm system. The Ubiquitous computing serves as a bone for such environmental improvement through the smart devices that are linked with Internet of Things (IoT). Considering the vast skyscraper cultivation with vast amounts of crop production, the storing, retrieving and updating of the information for the services is handled, to acquire precise agriculture benefits. For this data to be effectively used, it has to be transferred through multiple systems which results in energy loss and communication latency. In order to use the service in a decoupled way, we present RESTful Architecture style for an HTTP based service interface in the vertical farm system. We have also presented the integrated of REST API in both mobile and web application of the vertical farm monitoring system. Jersey is used for web services to simplify the RESTful service and client implementation.
Saraswathi Sivamani, Kyunghun Kwak, Yongyun Cho. **A Design of Web-based Services Using RESTful API for Vertical Farm.** *Life Sci. J* 2014;11(7):763-767] (ISSN: 1097-8135). http://www.lifesciencesite.com. 111

Keywords: Ubiquitous Computing; REST; u-Agriculture; IoT;

1. Introduction

Recently, the atmospheric changes and natural disaster are found in numbers, which results in major loss not only human life but also our atmosphere. Such as air, water and land are being polluted by nature. As a known fact the natural disasters are unpredictable. Vertical Farming [1], an indoor cultivation method has been considered a solution for this. Vertical farming helps in the growth of quality food production with the best atmospheric condition. The automated system reduces a considerable amount of manpower and increase productivity with the help of smart devices (sensors and actuators), which has been proved my many researches [2].

Due to the growth in ubiquitous environments, and with the Internet of things, the world has been completely merged with the Web that witnesses the improvement on a long scale. Thus, by incorporating the physical devices to the network, the information is used across the independent applications. More than one application can be built with the HTTP request - response. Rather than providing service, now we are able to access the services through the Open API Concept. Many major social sites such as Google, Yahoo, Facebook and Twitter are providing the Open API Services [3]. REST based service is most promising in terms of communication speed and integration of web services. REST being stateless reduces complexity, particularly in mobile based application.

Thus, we have used REST Web services in the Monitoring services which may help in the effective communication to perform CRUD operations. In this paper, we have proposed a design of monitoring services in the vertical farm with the HTTP based REST Architecture that can be integrated in both web and mobile application.

2. Background

2.1. Vertical Farming

As the economy trends out every day, the health conscious has become a major interest among people. Recent study proves that the interest in organic foods among the people have increased gradually [4], which was likewise a major reason greenhouse cultivation has become popular. But the production and reach of these products were limited. Vertical Farming helped to resolve the problem with indoor cultivation in high skyscrapers with the optimal environmental condition. In addition, Ubiquitous computing boosted the significance of agriculture among the consumer and providers [5]. Crop growth information and the conditions are received real time from the vertical farm that benefits and increase the buyers trust. With the REST API concepts applied to the system, the developer is given more opportunity to use the details in the user's application.

2.2. REST over SOAP

When we come across the OPEN API, there are many ways of approaching the Web Services such SOAP and REST based web services. SOAP (Simple Object Access Protocol) was the widely used Web Service still the REST (Representational State Transfer) was introduced. Although the REST [6] was introduced in the context of publishing hypermedia documents, it turned popular for the web services. Despite of the usage, SOAP and REST have their own cons and pros. The REST took over the SOAP, due to its better performance and scalability. It makes the user or developer to access the web service without any communication delay. It also helps to meet better scalability and performance. REST supports multiple format such as XML, JSON, HTML or whatever necessary, where SOAP is an XML based services. The Web service needs to step on two or more web services which may cause the communication latency. REST Web services help to minimize the communication delay more compared to the SOAP Web services. Overall the advantages of REST, integration of semantic web is the main advantages of REST, especially in the vertical farm as we are already using the Ontology based service model. As the context information and the REST access the resources by URI, the integration is flexible [7].

3. Vertical Farm Service Model with REST API

On a large-scale real time environment, an effective monitoring and controlling services of the vertical farm are realized through ubiquitous computing. To enrich the service on remote location through the web application, many studies are underway on the service oriented agricultural system [8-10]. Some of the research includes the farm management [11], disease diagnosis [12] and production sustainability [13].

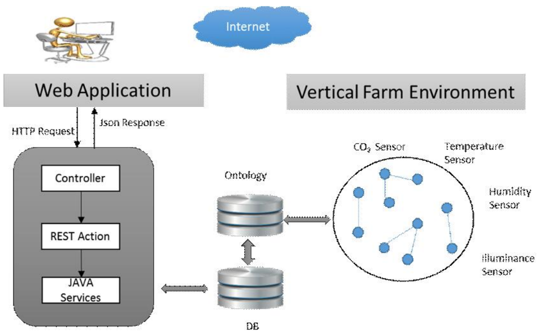

Figure 1. A REST based service flow Model

In our proposed system, the monitoring and controlling services are performed through the Web application which is implemented with REST architecture. In the vertical farming, all the sensed information such as temperature, humidity, illuminance and carbon-di-oxide from the sensor sink gets stored in the database. The real time sensed information from the database are retrieved and displayed in the web application. This information can be analyzed, stored and organized in a decoupled way with the REST API using the CRUD Operation.

Figure 1 describes a service model of the REST API communicating with the vertical farm system. Under the hood of the internet, the URI request is sent to the REST and obtain the response in the JSON format, which is used widely. REST API uses the Jersey implementation of JAX-RS, which facilitates in creating the network services in REST architectural style [14]. The sensed information are directed to the ontology where the context information are interpreted with the help of relationship and the web service discovery is performed. For developing the RESTful web services, Jersey framework is incorporated with the JAX-RS reference implementation. JAX-RS annotation makes it easy and faster to access the web services. Hence briefly reducing the communication latency which helps to improve the services in the vertical farm system.

As discussed earlier, the REST follows the URI pattern for the resources similar to ontology. In our previous study, we have explained the integration of the vertical farm ontology model which uses the OWL-S approach to represent the network services. The web services are linked to the OWL service model which is invoked through the HTTP request. Figure 2 shows the flow of sensed context information from the vertical farm to the Web application through the REST API.

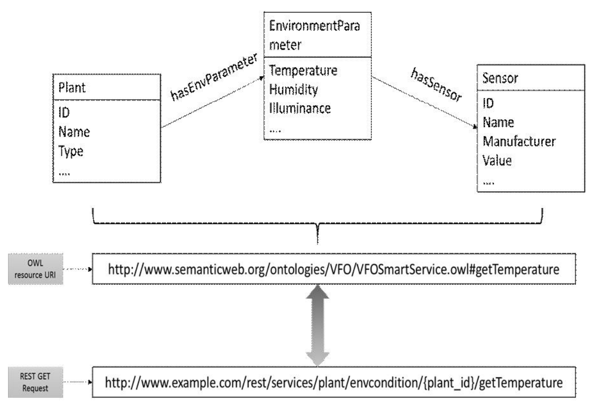

Figure 2. Mapping of ontology and REST

4. REST API Structure

In this section, we discuss the structure of the REST API used in the vertical farm system. A defined structure is more important for the developer which helps in the easy use and understanding of the API. Focusing the services in the vertical farm environment, the service is indicated in the start of the URI as shown in the Table 1.

Service is further divided into control equipment, sensor and location. Next the particular information is retrieved like sensorList, locationList, sensorId and etc. Finally the type of the response in mentioned in the URI. As we know, REST support many formats such as XML, JSON, HMTL and XHTML. We preferred using JSON as they are easily consumed by JavaScript and also in the human readable format.

Table 1. Structure of the URI

Service	Sensor	sensor_Id sensor_list Sensor_connected_list Sensor_waiting_list
	ControlEquip	Control_equipment_list Connected_list Waiting_list
	Location	Location_id Equip_location Sensor_location
	Plant	Grow Info List
	…	

The Uri pattern for the sensor information and the CRUD operation with the required parameter are shown in the Table 2.

Table 2. URI pattern for the sensor information

Request Type	URI
GET	service/sensor/connectedListDetails/{sensor_id}
GET	service/sensor/waitingListDetails/{sensor_id}
POST	service/sensor/add
PUT	service/sensor/modify/{sensor_id}
DELETE	service/sensor/delete/{sensor_id}

The defined structure makes it easier to process the data at different level separately. Referring the table 2, if you need to get the sensor list and the values, with the HTTP request on the connectedListDetails, all the information on the sensor and the data are retrieved. By parsing the retrieved JSON response, sensed data of the corresponding sensor is extracted.

5. Implementation

The design of the REST system that are to be considered before starting the RESTful web services are Resources, Resources class, URI and Representation. REST usually starts by identifying the objects and maps it to the resources, later by defining the identifier and operation on resources. A prototype web client was developed where the user is allowed to query the data on the monitoring system of vertical farming. The information on a particular plant can be retrieved, stored and updated where each plant was given a specific id. The growth information of the plant and their prediction can also be obtained together. Similarly the sensor, actuators, board, insect and other information can be directly accessed with the REST API.

Accordingly, a successful or a failure message is received along with the HTTP response. Although we have many OPEN API service in the vertical farm system, let us see some examples leaning to plant's information.

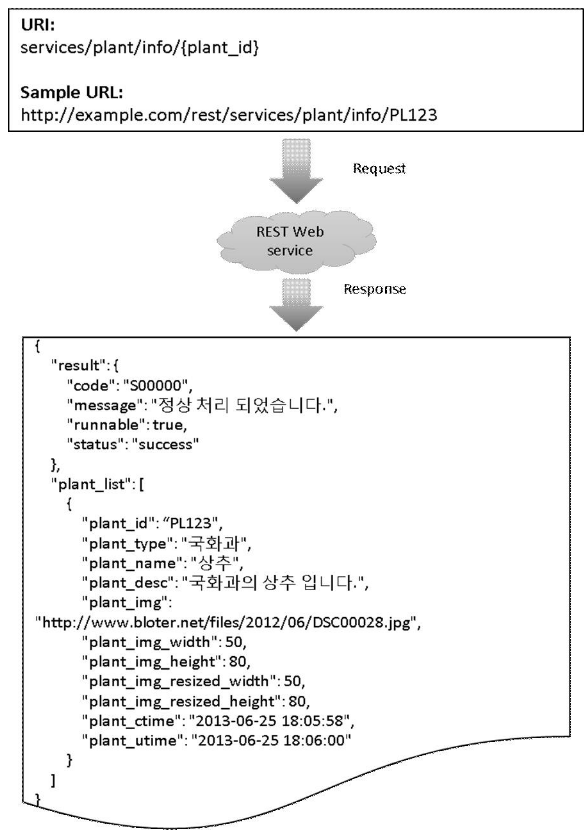

Figure 3. REST Web Service for the retrieval of Plant Information

Figure 3 shows an example of the REST web service to retrieve the information on a particular plant with the plant id. The HTTP GET request is used to retrieve the information. Figure 4 shows the example of REST web services to create a particular plant information with the required details. To create the resource, HTTP POST request is used. HTTP code 201 is returned for a successful response.

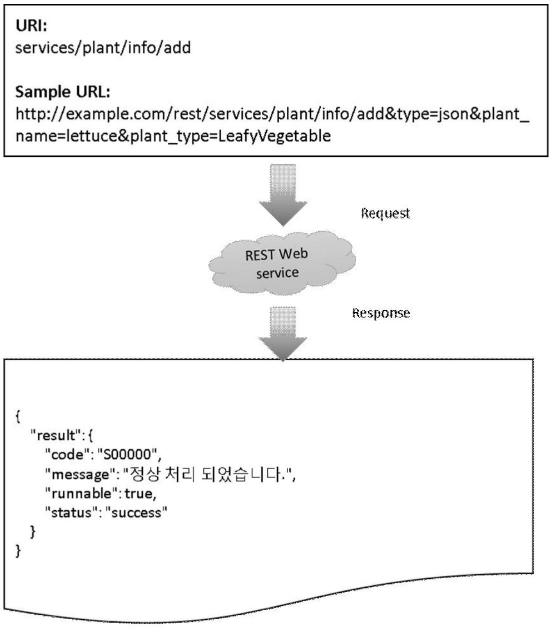

Figure 4. REST Web Service for the creation of Plant Information

Figure 5 shows the example of the error message for an unsuccessful REST web service. When performing an update operation (HTTP PUT), an error occurs as the given plant id does not exist. Thus returning the error message.

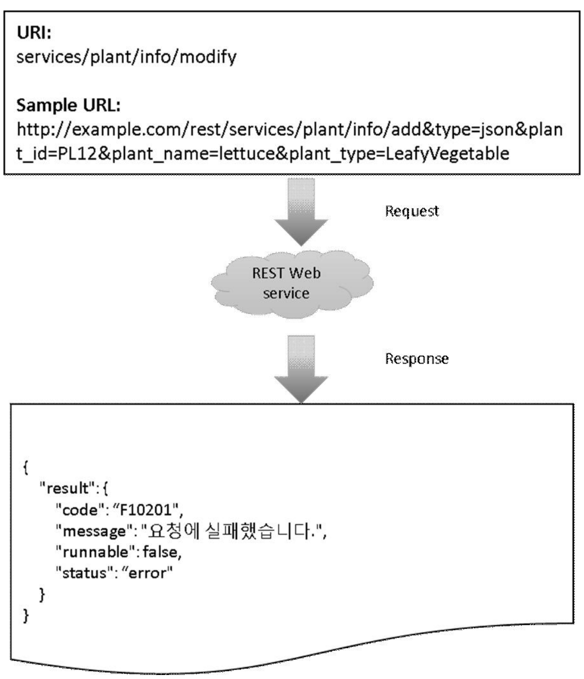

Figure 5. Error message in REST Web Services

6. Integration

REST API is commonly well known for its easy integration and test using HTTP request which can be performed with the web browser or poster plugin (available in Chrome and FireFox). An API that is developed for the Web application can also be used in the mobile application. REST is best suited for this kind as it is Stateless. Both Web and mobile application is created for the monitoring of the vertical farm system which invokes the REST API. The figure 6 shows the integration of REST API with both Web and Mobile Application.

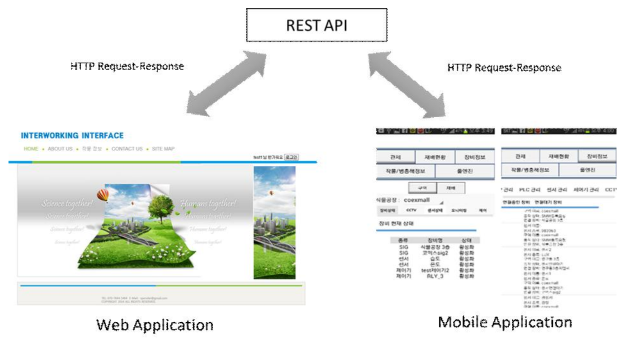

Figure 6. Integration of REST API in Vertical Farm Web and Mobile Application

The monitoring application is categorized into control, cultivation, Statistics, system management, environmental management and rule engine as shown in the figure 7.

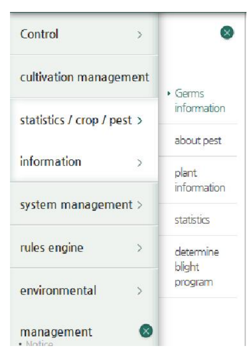

Figure 7. Vertical Farm Web Application Tab view

The control tab shows the state of the control equipment in each location and the connected sensor list. The Cultivation management deals with the plant information. Information regarding the pest, gem and the history of the plant is available in the statistics tab, where the growth prediction results are also obtained. Next tab contains the information on system such as sensor, actuators, etc. The environmental condition such as temperature, humidity and carbon-dioxide are included in the environmental management corresponding to each sector. The rule engine tab helps in the assignment of new schedules and crop stimulation process. The information is obtained by using the REST API which returns a JSON response.

7. Conclusion

In this paper, we have presented the REST based monitoring service for the vertical farm system. A prototype implementation was performed with both web and mobile application to identify the effectiveness of the REST web service for which the examples are discussed above. The proposed REST API helps in the decoupling of information system and share it in a human and machine readable format. It also helps in the storing, analyzing and retrieving of information effectively by reducing the communication latency. For the future work, we are planning the integration of the REST web services with semantic knowledge and implement an intelligent querying process.

Acknowledgement:

This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the CITRC (Convergence Information Technology Research Center) support program (NIPA-2013-H0401-13-2008) supervised by the NIPA (National IT Industry Promotion Agency).

Corresponding Author:
Prof. Yongyun Cho
Department of Information and Communication Engineering
Sunchon National University.
Suncheon, Jeonnam 540-742, S.Korea
E-mail: yycho@sunchon.ac.kr

References

1. Besthorn, F. H. (2013). Vertical farming: Social work and sustainable urban agriculture in an age of global food crises. Australian Social Work, 66(2), 187-203.
2. Zhou, L., Song, L., Xie, C., & Zhang, J. (2013). Applications of Internet of Things in the Facility Agriculture. In Computer and Computing Technologies in Agriculture VI (pp. 297-303). Springer Berlin Heidelberg.
3. Ko, M. N., Cheek, G. P., Shehab, M., & Sandhu, R. (2010). Social-networks connect services. Computer, 43(8), 37-43.
4. Hjelmar, U. (2011). Consumers' purchase of organic food products. A matter of convenience and reflexive practices. Appetite, 56(2), 336-344.
5. Cho, Y., Cho, K., Shin, C., Park, J., & Lee, E. S. (2012). An agricultural expert cloud for a smart farm. In Future Information Technology, Application, and Service (pp. 657-662). Springer Netherlands.
6. Webber, J., Parastatidis, S., & Robinson, I. (2010). REST in Practice: Hypermedia and Systems Architecture. O'Reilly Media, Inc.
7. Battle, R., & Benson, E. (2008). Bridging the semantic Web and Web 2.0 with representational state transfer (REST). Web Semantics: Science, Services and Agents on the World Wide Web, 6(1), 61-69.
8. Sivamani, S., Bae, N., & Cho, Y. (2013). A Smart Service Model Based on Ubiquitous Sensor Networks Using Vertical Farm Ontology. International Journal of Distributed Sensor Networks, 2013.
9. Xia, J., Tang, Z., Shi, X., Fan, L., & Li, H. (2011, December). An environment monitoring system for precise agriculture based on wireless sensor networks. InMobile Ad-hoc and Sensor Networks (MSN), 2011 Seventh International Conference on (pp. 28-35). IEEE.
10. Cho, Y., Park, S., Lee, J., & Moon, J. (2011). An OWL-based context model for U-agricultural environments. In Computational Science and Its Applications-ICCSA 2011 (pp. 452-461). Springer Berlin Heidelberg.
11. Nikkilä, R., Seilonen, I., & Koskinen, K. (2010). Software architecture for farm management information systems in precision agriculture. Computers and electronics in agriculture, 70(2), 328-336.
12. Manhire, J., Moller, H., Barber, A., Saunders, C., MacLeod, C., Rosin, C., & Barr, T. (2012). The New Zealand sustainability dashboard: unified monitoring and learning for sustainable agriculture in New Zealand.
13. Kolhe, S., Kamal, R., Saini, H. S., & Gupta, G. K. (2011). A web-based intelligent disease-diagnosis system using a new fuzzy-logic based approach for drawing the inferences in crops. Computers and Electronics in Agriculture, 76(1), 16-27.
14. Hadley, M., Pericas-Geertsen, S., & Sandoz, P. (2010, April). Exploring hypermedia support in Jersey. In Proceedings of the First International Workshop on RESTful Design (pp. 10-14). ACM.

5/26/2014