Available online at www.sciencedirect.com

ScienceDirect

Systems and Software 00 (2016) 1–3

Systems and Software

A Holistic IoT User-Friendly Management System

Claudio A. S. Wunder, Caio C. M. Silva, Rafael T. S. JR

Abstract

Keywords:

1. Introduction

2. Related Concepts

3. Proposal

In this section it is presented the IoT management system approach. Figure 1 presents the proposed architecture. The system architecture was conceived into five layers, denominated, user experience layer (UXL), presentation layer (PL), input interpreter layer (IIL), business logic layer (BLL), and communication layer (CL).

Broadly speaking, the UXL was idealized to make possible an IoT management through a natural interface. In the other hand, the PL receive a HTTP request, which represents a solicitation of a system feature, and routes to the respective feature handler. The IIL digest and solve the received data from PL and invoke the respective system action. The BLL receive the invoked action from IIL and execute the solution to complete the required feature. The CL does the communication using a REST approach as indicated in [1], sending requests to a service layer.

The proposal presentation is divided into four subsections, where each section describes each architecture layer of the proposed system.

3.1. Presentation Layer

The presentation layer provides a full aware IoT network management context.

3.1.1. Routing Module

3.1.2. Rendering Module

3.2. Input Interpreter Layer

The IIL is liable to understand the requested resource. In order to interpret the submitted request, IIL performs the process presented in Figure 2.

The figure process starts when PL sends a resource request to IIL. The request data is received by Input Solver which solves and map it.

In the next step, a raw data map is sent to Input Digester, which cleans the data, removing useless and uncomprehensible information. If the map does not represent a valid set of information, the data is discarded and the process ends. In other way, the map is redirect to Input Interpreter which associate the data map elements into valid resource elements. Finally, the map is treated in Input Transceiver, joining and associating similar resources.

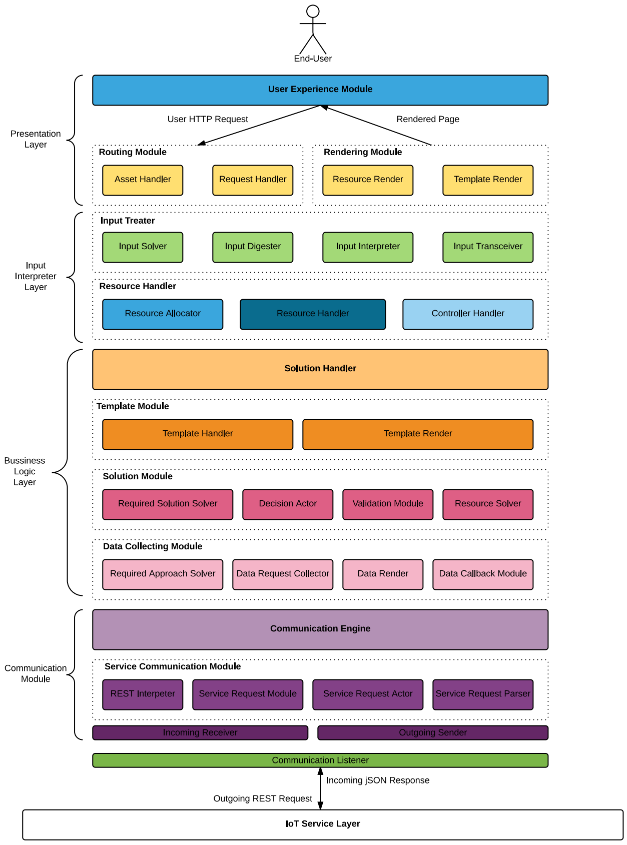

Figure 1. IoT management system architecture.

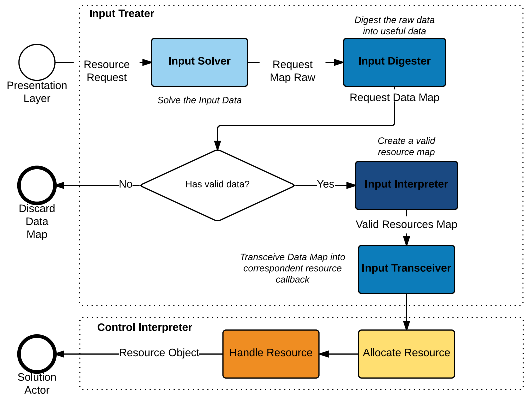

Figure 2. Input Interpreter Layer Flow Chart

Thus, the resource map is routed to Resource Handler Module, which allocate the specific resources into memory. In addition, the resource handler performs all invocations related to it correspondent resource.

The following sub sections intents to explain each component of input interpreter layer, presenting their respective functionalities and limitations.

3.2.1. Input Treater

The IT has four components, input solver, input digester, input interpreter and input transceiver. The input solver and digester use predefined regular expressions to identify which input expressions are valid. Those validated expressions are assigned to a correspondent route.

For an example, the Listing 1 shows a route being created. The function *addRoute* receives five parameters. The first one indicates the allowed regular expression for the route. The second one, set the type of node that will be invoked. Moreover, the third one defines the route segment level. Finally, also is defined the RESTful method, and the public route name.

Listing 1. Route Assign

```
Router::addRoute(’/(\w+)’, new ActionNode
    , 2, ’get’, ’web_page’);
```

After the route assign, the input interpreter receives the resource map, which identifies the correspondent resource callback for each data, ignoring information that does not belong to any existing resource. Thereby, a full comprehensible resource map is obtained, containing valid resource information. The resource map elements are simplified by input transceiver, which joins resources of the same type. Finally, the resource map is sent to the resource handling layer, which is explained in Section 3.2.2.

3.2.2. Resource Handler

3.3. Business Logic Layer

3.3.1. Solution Handler

3.3.2. Template Module

3.3.3. Solution Module

3.3.4. Data Collecting Module

3.4. Communication Module

3.4.1. Communication Engine

3.4.2. Service Communication Module

3.5. Communication Interface

Table 1. *Caption* comes before the table.

	title page	odd page	even page
onesided	leftTEXT	leftTEXT	leftTEXT
twosided	leftTEXT	rightTEXT	leftTEXT

3.6. User Experience

4. Experimental Environment and Result Analysis

5. Conclusion and Future Work

Acknowledgments

The authors wish to thank the Brazilian research, development and innovation Agencies CAPES (Grant FORTE 23038.007604/2014-69), FINEP (Grant RENASIC/PROTO 01.12.0555.00) and the National Post-Doctorate Program (PNPD/CAPES) for their support to this work.

References

[1] C. C. de Melo Silva, H. G. C. Ferreira, R. T. de Sousa Júnior, F. Buiati, L. J. G. Villalba, Design and evaluation of a services interface for the internet of things, Wireless Personal Communications 1–38.