

# On the Creation of Customizable Laboratory Experiments: Deconstruction of Remote Laboratories to Create Laboratories as a Service (LaaS)

<http://dx.doi.org/10.3991/ijoe.v10i6.3989>

A. C. Caminero, A. Robles-Gómez, S. Ros, Ll. Tobarra, R. Hernandez, R. Pastor, and M. Castro  
Universidad Nacional de Educación a Distancia, UNED, Madrid, Spain

**Abstract**—In the recent years, the research community has been focused on cloud systems as a way to provide customizable on demand access to computing resources in a variety of flavors (the most widely known being Infrastructure as a Service, IaaS, Platform as a Service, PaaS, and Software as a Service, SaaS). In this work, we strive to provide our vision on cloud-based laboratories, which allow the on-demand creation and customization of experiments, these being called Laboratories as a Service (LaaS). Our LaaS will be based on the deconstruction of remote laboratories, creation of clients, and selection of a container. An aeolian laboratory based on the Lego Mindstorms robotic kit is used as an example.

**Index Terms**—Remote laboratories, Distance education, Laboratories as a Service (LaaS), Customization.

## I. INTRODUCTION

The research community has been focusing recently on the provision of different utilities as services – these being called Anything as a Service (XaaS) [1]. Some of the more commonly used utilities are Infrastructure as a Service (IaaS) such as Amazon Elastic Compute Cloud [2], Platform as a Service (PaaS) such as Google App Engine [3], and Software as a Service (SaaS) such as Google Docs [4].

In parallel, the educational community, more precisely, those working on distance learning, has been focusing on the provision of remote laboratories to allow our distant students access them as if they were physically in the same location as the laboratory. For this, a Remote Laboratory Management System (RLMS) is needed, such as WebLab Deusto [5], LabShare Sahara [6], Related [7], or LiLa [8]. Remote laboratories are an essential tool for distance education, as our case is, especially in the field of engineering, where hands-on experience in the use of equipment is one of the main assets of students once they finish their degree and start their professional careers.

In our case, the National University for Distance Education of Spain (Universidad Nacional de Educación a Distancia, UNED), is the largest university in Spain, with more than 200,000 students. We provide totally distant education, so the use of remote laboratories to obtain practical knowledge on technical topics becomes a top necessity.

Taking into account all of this, we propose the creation of a service-based utility that allows remote laboratories to be consumed from third parties in a versatile and customizable manner, this being called Laboratories as a Service (LaaS). These LaaS are based on the deconstruction [9] of remote laboratories into smaller “chunks”, which can be combined in order to allow the creation of customizable remote laboratory experiments. Thanks to this, users of remote laboratories (e.g. faculty) can create experiments that fit into their needs, and avoid unnecessary functionalities that may jeopardize the acquisition of knowledge by students. This way, the same laboratory can be adapted to the needs of its users, increasing the usefulness of the system – and perhaps also its revenue.

The development of LaaS has among others the following benefits. LaaS allow the versatile creation of on-demand experiments. This can be achieved by means of using the laboratory services that are needed for each specific experiment, avoiding the need to use all the services that are not needed, and thus keeping students away from the unnecessary complexity which may jeopardize their learning process. For instance, one single remote laboratory can be used for experiments in a basic course, but also for experiments in advanced courses: for a basic-level experiment, only a small subset of the laboratory services could be used, which isolates students from the complexity of the laboratory which is unnecessary (and perhaps even out of their scope) at this level. Then, for an advanced level course, experiments can be created by using a wider set of services, which allow students have more influence and control over the experiments and their results. Thanks to this, the same laboratories can be used for a wide set of experiments (just varying the services that are visible in each case), which increases their levels of use. This also affects the revenue of the institution developing the laboratories, since they can be used for more purposes at virtually the same cost. The creation of LaaS includes several points. First, remote laboratories must be deconstructed, this is, their functionalities must be provided independently and in a service-basis (similarly to [9]). Second, clients consuming the laboratories services must be created, similarly to [10]. Finally, a container must be chosen in order to hold these clients and create customizable experiments, similarly to [10].

This work presents the description of our work on the development of LaaS. The structure of this paper is as

follows: Section II describes the motivations for our work; Section III describes the features of the laboratories used in this work; Section IV details our work on the creation of LaaS; and Section V presents conclusions and suggests guidelines for future work.

## II. MOTIVATION

The Spanish University for Distance Education (UNED, Universidad Nacional de Educación a Distancia) [11] is a distance university devoted to provide university training to students who lack the time or resources needed to attend on-campus classes. This is necessary for students who have tight timetable because of other commitments (e.g. work or family commitments), thus fitting on-campus classes into their timetable becomes unfeasible. Furthermore, UNED provides university training to convicts and sick people, whose attendance to on-campus classes is not possible. UNED provides such students with the platform to receive education at home without the time-consuming and sometimes money-consuming effort requested by on-campus universities. The evolution of education and the increase in the knowledge necessities our society requires have created significant changes with regard to the way how the learning process takes place. Nowadays, there is a constant need to improve, to keep our knowledge up-to-date or to obtain knowledge on new topics – this being specially true in the case of technical studies, where technology is constantly evolving. Distance education is a solution to this problem, since it allows students to obtain practical knowledge without the space and time constraints of classical face-to-face education – thus allowing them to fit their studies into their possibly tight schedules.

Courses at UNED are managed at the central offices in Madrid, where lecturers attend students from anywhere by means of the technological infrastructure. UNED has developed its own Learning Management System (LMS), called aLF [12], which is based on dotLRN [13]. The technological infrastructures more commonly used are email and forums, that are used to allow students to virtually meet each other and lecturers and simulate the interactions that take place in traditional face-to-face classes.

UNED employs local tutors, who are spread over a number of associated centers all over Spain. These tutors provide the only on-campus access to students, which allow them to ask questions personally. Besides, associated centers allow students to use academic resources (e.g. books). For subjects with few students enrolled (e.g. subjects in the last courses of engineering degrees), some associated centers may decide not to provide tutors. In this case, UNED provides webconference tutorial sessions, which allow students without tutorial sessions

in their local associated center for a given subject, access tutorial sessions online for that subject. These webconferences allow synchronous interchange of ideas and comments between tutors and students. Besides, webconferences can be recorded and published in the subject's virtual course in aLF. Even more, recordings can be downloaded in mp4 format, so that students can view them offline.

It can be seen the tremendous necessities of our University with regard to the distant access to the teach-

ing resources, in which laboratories are a key. This highlights our need to provide distant laboratories. Furthermore, the creation of LaaS is a really interesting issue with regard to the re-usability and versatility of our laboratories.

## III. DESIGNING LABORATORIES

There are different laboratories developed by commercial companies. In particular, there are several companies dedicated to renewable energy. Among others, we can find Alecop [14], Lucas Nuelle [15], Edibom [16], and Elektron [17]. They provide kits to create several types of renewable energy laboratories, such as aeolian or solar. Apart from laboratories, they provide didactic guides that ease the user's learning process. These laboratories are thought to be used in face-to-face education where students access the equipment physically, or using a desktop application within a local network. For these reasons, they are not suitable at distance education, as this case is. In this sense, there is a need to choose a platform for designing low-cost remote laboratories to be employed through the Internet, without face-to-face interactions.

In the recent years, several platforms to construct low cost laboratories have emerged, such as Arduino or Raspberry Pi. These are simple platforms, but capable of achieving very complex developments. Their main drawback is that once the laboratory is designed and built, hardware components must be soldered, which makes difficult the reuse of laboratory elements in order to create new laboratories. Considering this, the Lego Mindstorms NXT v2.0 [18] kit came to mind. Its modular design allows the easy creation and modification of laboratories, which can be connected to a PC using a USB cable or a Bluetooth connection. Besides, a variety of tools and libraries are available to be used with Lego Mindstorms. Since the Lego kit can be assembled and disassembled as many times as needed, the educational activities can easily be changed or modified. As an example, the aeolian laboratory can be transformed into a solar laboratory using the same pieces. Also, the laboratory design can be adapted to new needs without purchasing new hardware components. The maintenance is also quite simple, since any broken or lost piece is easily replaceable at a low cost. These reasons suggest that our LaaS should be based on Lego Mindstorms NXT v2.0 and its renewable energy kit [19]. Each robot/experiment

represents an investment of up to five hundred euros.

Using the renewable energy kit implies the use of a new sensor for the Lego robot, called Energy Meter. This sensor is the responsible for storing the power generated from a dynamo motor or a small solar panel (both included in the same kit).

To build the aeolian laboratory, some blades are connected to the dynamo motor, which will generate electric energy that is stored in the Energy Meter sensor. Figure 1 shows a schema of the aeolian robot. Similarly, the solar laboratory is built connecting the solar panel to the Energy Meter sensor, and this sensor to the Lego NXT v2.0 block. The Energy Meter sensor also measures a set of parameters related to the power production (such as voltage, electric current, joules stored in the battery, and watt), as well as storing the electrical power. Figure 2 shows a schema of the solar robot.

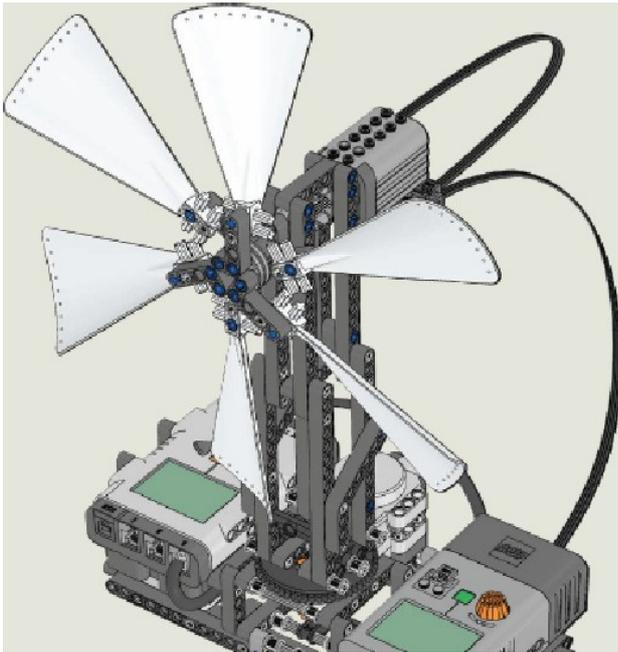


Figure 1. Aeolian robot.

#### IV. CREATING LAAS

Several ways to do this can be used, for instance, by means of using Mashups [20], [21], which allow easy generation of personalized environments. Two points must be considered which define future functionalities of such environments. These points are (1) container selection [22], [23], [24], which allows the deployment of different gadgets, and (2) the architecture selection [25].

As said before, creating LaaS requires three steps. In the first place, laboratories must be deconstructed (similarly to [9]), which means that the functionalities of the laboratory are split so that they can be rearranged in order to create customized experiments. Next, client applications that use the aforementioned functionalities must be created (similarly to [10]). Finally, a container to host these applications and publish them for its use must be selected. These steps along with their outcomes are outlined in Figure 3, and are detailed in this section.

This section details these steps and provides examples on how they have been applied to a laboratory on renewable energy.

##### A. Step 1. Deconstructing laboratories

Deconstructing laboratories allows the creation of basic building blocks, instead of full-featured laboratory experiments. Thanks to this, the laboratory functionalities are separated from each other, and some of them (chosen by faculty) can be arranged again in order to create customizable experiments by using only the needed functionality. This way, faculty can create customized experiments on demand, which fit the actual needs of the learning outcomes that are pursued in this experiment, avoiding the unneeded functionalities which may jeopardize the learning process of students.

For this work, a laboratory on renewable energy has been deconstructed. This laboratory is based on the Lego Mindstorms NXT v2 robotic toolkit [18] extended with renewable energy kit [26] which uses a sensor called Energy Meter. The Energy Meter sensor can measure

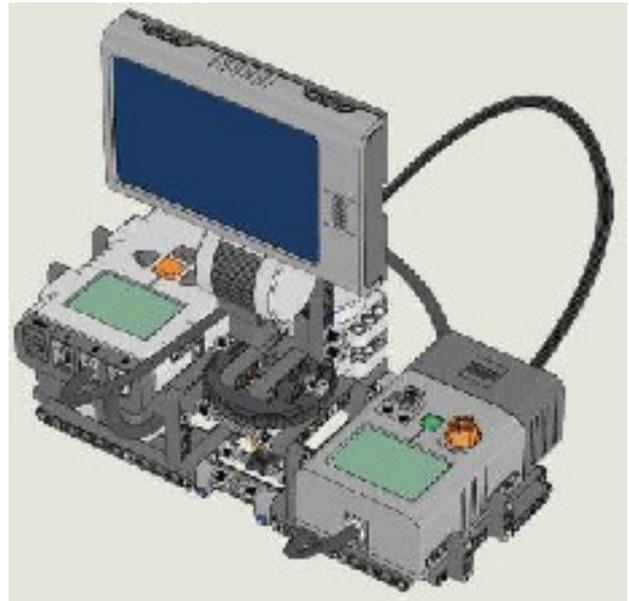


Figure 2. Solar robot.

a set of parameters related to the power production (such as voltage, electric current, joules stored in the battery, and watt), as well as storing the electrical power. This way, a remote lab on aeolian energy is implemented. This aeolian laboratory focuses on helping students to understand how wind is directly related to the generation of energy. For this reason, students are asked to register and observe the generation of voltage for different wind power values. A picture of the aeolian laboratory can be seen in Figure 1. Figure 4 shows the proposed architecture for this work.

Figure 5 shows the original remote laboratory interface, which is a web interface based on HTML5, jQuery, and the Bootstrap library. This approach allows us to use the laboratory with a wide range of devices and operating systems. We have tested the functionality of our laboratory with Firefox and Chrome browsers, and Windows, Linux, Android (mobile and tablet), and iOS for iPad devices.

This web interface is used to start and end the experiments, temporarily store the generated data, and export the data to be processed by the student in CSV format. Also, it includes a graphical plot from the data captured by the Energy Meter sensor, in order to show the values of each of the possible variables. The actuator of the laboratories can be controlled by using a slider located at the left side of the page, as observed in Figure 5. Furthermore, video images from a webcam placed next to the robots are also included in the web site, providing students with a visual feedback of what happens in the laboratory. In order to help students during an experiment session, several aid messages have been included in the website, that pop-up when they hover the mouse over the controls and web elements. These messages strive at explaining the meaning and operation of the laboratory. A button has also been added to the site in order to display modal windows with information on a particular activity that the student must perform and how the laboratory works. This button is available to the student during the whole experiment.

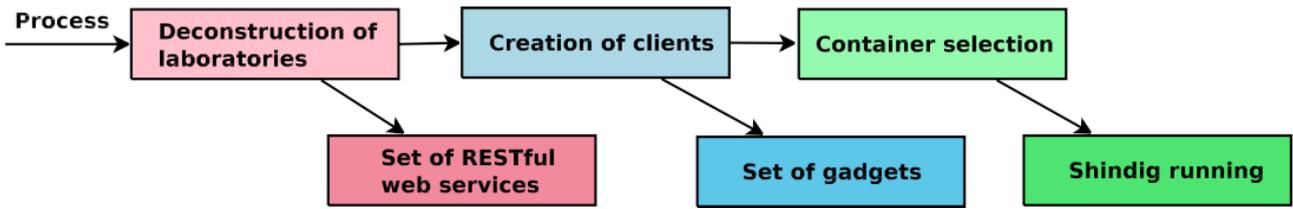


Figure 3. Process steps and outcomes.

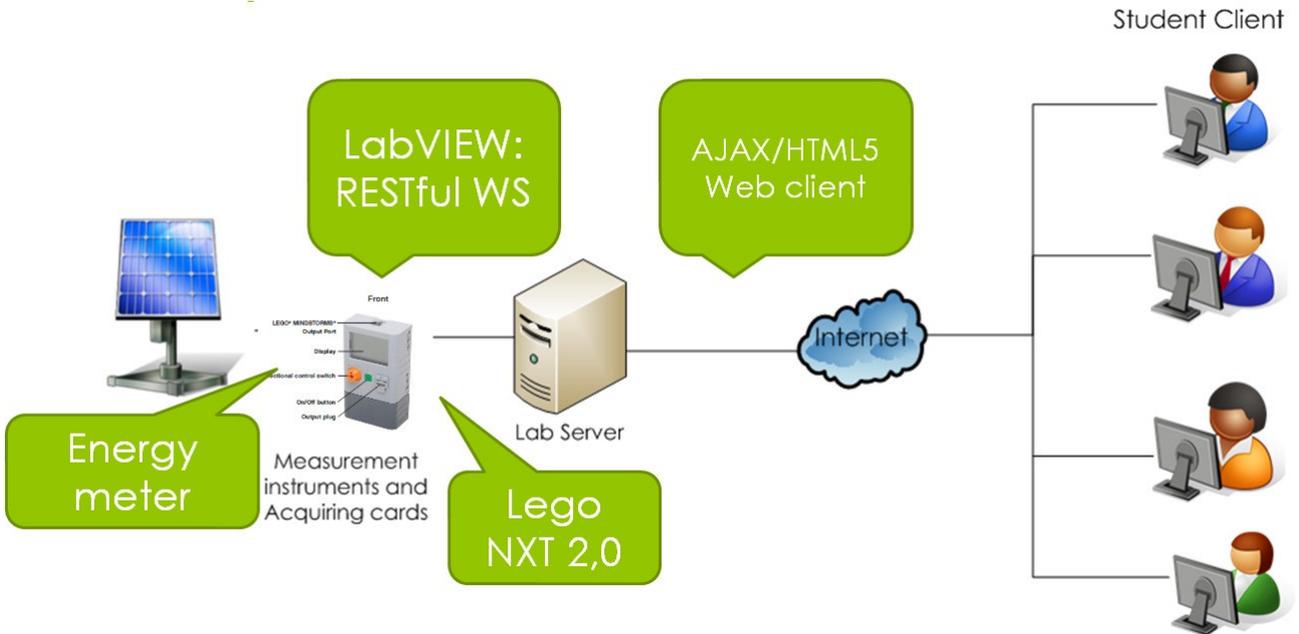


Figure 4. Structure of the low cost remote laboratory.

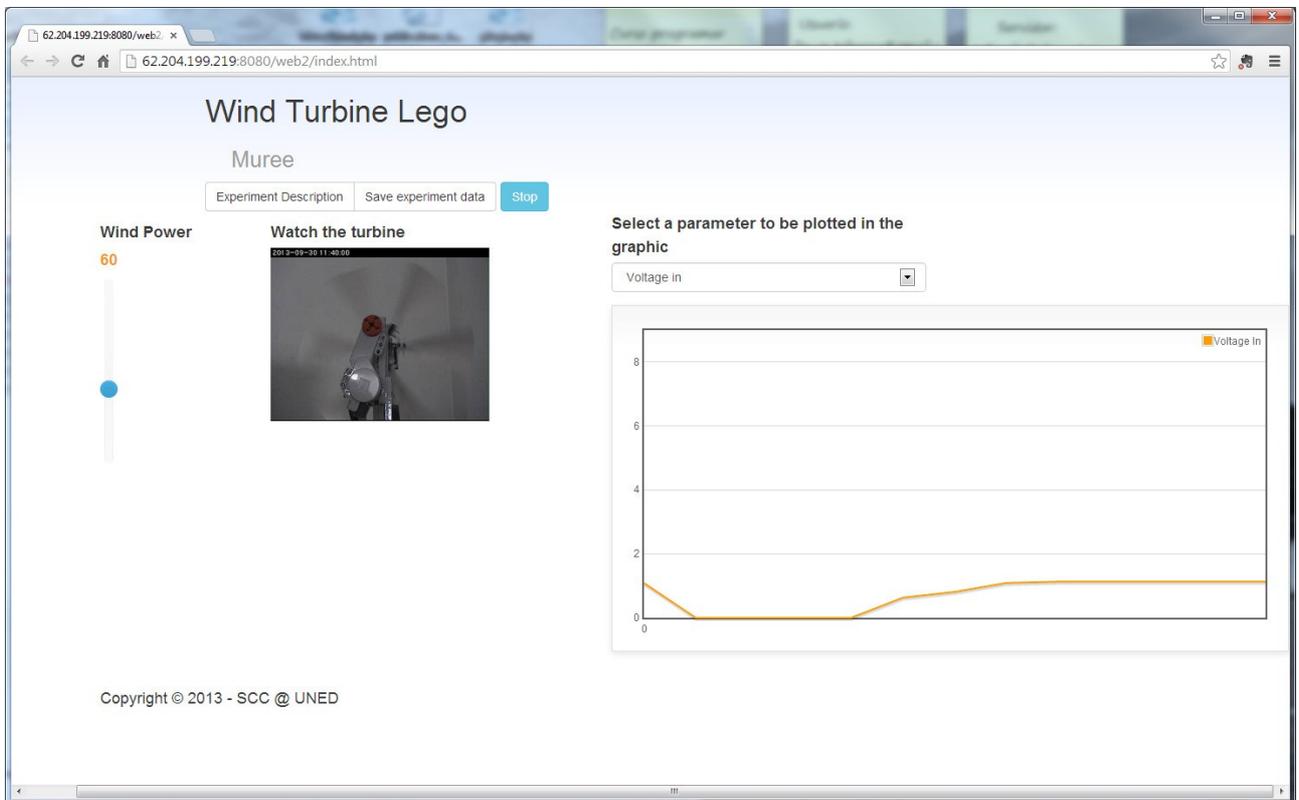


Figure 5. Remote laboratory's web interface.

Taking all of this into account, we have developed a number of RESTful web services using LabVIEW, which is a novel approach that allows users to easily create multiplatform laboratories. These services provide fine-grained functionalities, that can be better organized in order to allow the on-demand creation of customized laboratory experiments. The services we provide initially include checking the status of the laboratory, plotting data graphs, and modifying input data. These services map the functionalities available in the former web interface of the laboratory presented in Figure 5.

### B. Step 2. Creating clients

Client applications are implemented as gadgets, which are mini-applications that can be assembled by different OpenSocial-compliant providers. Gadgets are portable, can be customized, and support federated security through OAuth.

Taking into account that gadgets are inherently modular, a number of them have been created to consume the services published by the laboratory. The functionalities gadgets provide include checking the status of the laboratory, plotting a graph, or modifying the actuators of the laboratory. An excerpt of the JavaScript code for a gadget which returns the status (ok or failure) of a laboratory is presented in Table I. This code is based on OpenSocial [27] gadget's API and uses JavaScript Object Notation (JSON) to query laboratories. This format has been chosen for its simplicity and better performance.

OpenSocial is an open set of programming interfaces (APIs) for web-based social network applications. Originally pioneered by Google, the OpenSocial API covers a broad range of capabilities and functionality, including Profiles, Relationships, Activities, Shared applications, Authentication and Authorisation. The primary goal of OpenSocial is to provide a common framework for developers who want to ensure interoperability across various social networks on the Internet, which act as containers for each OpenSocial-compliant application. OpenSocial includes multiple APIs for social software applications to access data and core functions on participating social networks. OpenSocial allows the construction of many types of gadgets, for instance, gadgets that run within a calendar and display time-based information, gadgets that trigger based on email context, or gadgets that interact with the content of a spreadsheet.

### C. Step 3. Selecting a container

A container is needed to use the gadgets created for each service of a laboratory. In our case, Apache Shindig [28] has been chosen as container for our gadgets for several reasons. First, this is the reference implementation of the OpenSocial standard. Besides, several major users of Shindig include [29] LinkedIn [30], Hi5 [31] or Yahoo [32]. Even more, learning tools such as Graasp [33] are based on Shindig to provide e-learning services.

The primary goal of Apache Shindig is to provide infrastructure to host OpenSocial apps on websites. Another goal of Apache Shindig is to be language neutral

and cover multiple languages. Currently, Java and PHP versions are available and supported.

Apache Shindig has a number of interesting functionalities, including:

- Gadget Container JavaScript: core JavaScript foundation for general gadget functionality. This JavaScript manages security, communication, UI layout, and feature extensions, such as the OpenSocial API.
- Gadget Rendering Server: used to render the gadget XML into JavaScript and HTML for the container to expose via the container JavaScript.
- OpenSocial Container JavaScript: JavaScript environment that sits on top of the Gadget Container JavaScript and provides OpenSocial specific functionality (profiles, friends, activities, datastore).
- OpenSocial Data Server: an implementation of the server interface to container-specific information, including the OpenSocial REST APIs, with clear extension points so others can connect it to their own backends.

An installation of Shindig based on a XAMPP container has been performed, in which our gadgets have been tested. A XAMPP container has been used since it is an easy way to have the server up and running.

Figures 6 and 7 present two screenshots of our deconstructed laboratory. Figure 7 presents the output of the gadget presented in Table I, as part of an Apache Shindig server. Figure 6 presents a testing Shindig server where two gadgets (the lab status and the Wind Power slider) are depicted.

## V. CONCLUSIONS AND FUTURE WORK

The creation of customizable laboratory experiments is a very useful asset for engineering studies, since the same laboratories can be used for a wide set of experiments, which increases their levels of use and even their revenue. In this paper we present our work on the creation of customizable laboratory experiments, also known as Laboratories as a Service, (LaaS). These are based on the deconstruction [9] of remote laboratories into smaller parts that can be arranged separately depending on the teachers' needs. This way, several improvements can be achieved, in terms of better use of existing laboratories, better learning results, reduced learning curve, or higher revenue of remote laboratories.

We identify three steps to create LaaS, these being (1) deconstruction of laboratories (based on RESTful web services, similarly to [9]), (2) creation of clients (based on OpenSocial gadget API, similarly to [10]), and (3) selection of a container (supporting OpenSocial, similarly to [10]). These steps are detailed in this work, along with examples on how they have been performed in an aeolian laboratory and an solar laboratory based on the Lego Mindstorms robotic kit.

Among our future work, we plan to create a full LaaS, based on the deconstruction of the aeolian laboratory. After that, this LaaS is going to be used in subjects of engineering education and assess its benefits.

TABLE I.  
CODE SNIPPET FOR THE LAB STATUS GADGET.

```

<?xml version="1.0" encoding="UTF-8"?>
<Module>
  <ModulePrefs title="Status" description="Remote Laboratory
    Status" height="250" width="400"/>
  <Require feature="opensocial-0.9"/>
  <Require feature="osapi"/>
  <Content type="html" view="canvas,home">
    <![CDATA[
      <link rel="stylesheet"
        href="http://62.204.199.229/gadgets/bootstrap/css/bootstrap.css
        " />
      <script language="javascript" type="text/javascript"
        src="http://62.204.199.229/gadgets/jquery-1.10.2.min.js"></script>
      <script language="javascript" type="text/javascript"
        src="http://62.204.199.229/gadgets/bootstrap/js/bootstrap.js"></scr
        ipt>
      <script language="javascript" type="text/javascript"
        src="http://62.204.199.229/gadgets/jquery.base64.js"></scr
        ipt>
      <script language="javascript" type="text/javascript"
        src="http://open-
        app.googlecode.com/files/openapp.js"></script>
      <h2> Laboratory current status </h2>
      <div id="labstatus"><div>
      <script type="text/javascript">
        function makeJSONrequest(){ var params = {};
          params[gadgets.io.RequestParameters.CONTENT_TYPE] =
            gadgets.io.ContentType.JSON;
          params[gadgets.io.RequestParameters.METHOD] =
            gadgets.io.MethodType.GET;
          var
            url="http://62.204.199.219:8080/WindTurbine2/Status"
            gadgets.io.makeRequest(url, callback, params);
        }
      function callback(obj){ console.log("Callback...");
        if(obj.rc>399){
          var cleanjson=obj.text.replace(/(\r\n|\n|\r)/gm, ". ");
          var json=gadgets.json.parse(cleanjson);
          var text="<p> There is a problem with the remote laborato-
            ry." + " Please contact your laboratory administrator.
            </p>";
          text+="<p> Error code: "+ obj.rc+"</p>";
          text+="<p> Error description : "+json.Status[3]+"</p>";
          $('#labstatus').html("<div class='alert alert-danger'>"+
            text+"</div>");
        }
        else $('#labstatus').html("<div class='alert alert-
            info'> The remote laboratory is ok. </div>");
      }
    ]]>
    console.log("Loading..."); gadg-
    ets.util.registerOnLoadHandler(makeJSONrequest); con-
    sole.log("Loaded!!!");
      </script>
    </Content>
  </Module>

```

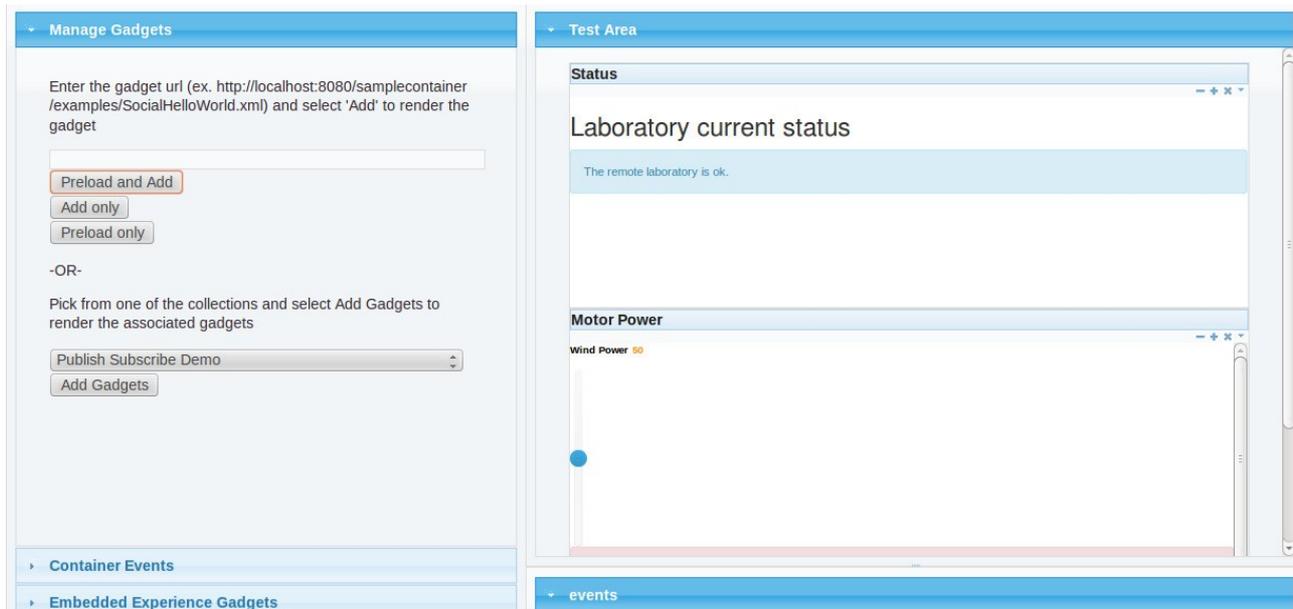


Figure 6. Gadgets running on a testing Shindig server.

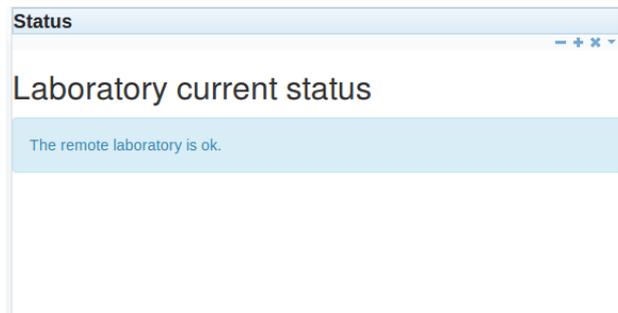


Figure 7. Graphical output of the lab status gadget.

## REFERENCES

- [1] Y. Duan, "Value modeling and calculation for everything as a service (XaaS) based on reuse," in 13th Intl. Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Kyoto, Japan, 2012.
- [2] Amazon Elastic Compute Cloud, Web page at <http://aws.amazon.com/ec2/>, Date of last access: June 30, 2014.
- [3] Google App Engine, Web page at <https://developers.google.com/appengine/>, Date of last access: June 30, 2014.
- [4] Google Docs, Web page at <https://docs.google.com/>, Date of last access: June 30, 2014.
- [5] WebLab Deusto, Web page at <https://www.weblab.deusto.es/web/>, Date of last access: June 30, 2014.
- [6] LabShare Sahara, Web page at <http://sourceforge.net/projects/labshare-sahara/>, Date of last access: June 30, 2014.
- [7] R. Pastor, D. Sánchez, S. Ros, R. Hernández, A. C. Caminero, L. Tobarra, A. Robles-Gómez, M. Castro, G. Díaz, E. S. Cristóbal, and M. Tawfik, "An xml modular approach in the building of remote labs by students: A way to improve learning," Intl. Journal of Online Engineering (iJOE), vol. 9, no. S5, pp. 5–12, 2013. <http://dx.doi.org/10.3991/ijoe.v9iS5.2662>
- [8] Library of labs, LiLa, Web page at <http://www.lila-project.org/>, Date of last access: June 30, 2014.
- [9] S. Ros, A. C. Caminero, A. Robles-Gómez, R. Hernández, R. Pastor, T. Read, A. Pesquera, and R. Muñoz, "Deconstructing VLEs to create customized PLEs," in Proc. of the 16th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITiCSE), Darmstadt, Germany, 2011.
- [10] S. Ros, R. Hernández, A. Robles-Gómez, A. C. Caminero, L. Tobarra, and E. S. Cristóbal, "Open service-oriented platforms for personal learning environments," IEEE Internet Computing, vol. 17, no. 4, pp. 26–31, 2013. <http://dx.doi.org/10.1109/MIC.2013.73>
- [11] Universidad Nacional de Educación a Distancia (UNED), Web page at <http://www.uned.es/>, Date of last access: June 30, 2014.
- [12] R. Pastor, T. Read, S. Ros, R. Hernández, and R. Hernández, "Virtual communities adapted to the EHEA in an enterprise distance e-learning based environment," in Proc. of the Third Intl. Conference on Online Communities and Social Computing (OCSC), held as part of 13th Intl. Conference on Human-Computer Interaction (HCI), San Diego, USA, 2009.
- [13] .LRN, Web page at <http://www.dotlrn.org/>, Date of last access: June 30, 2014.
- [14] Alecop, Web page at <http://www.alecop.com/equipment/index.php/es/equipamiento-didactico/energias-renovables-y-eficiencia-energetica>, Date of last access: June 30, 2014.
- [15] L. Nuelle, Web page at <http://www.lucasnuelle.com/316/apg/1225/Renewable+Energies.htm>, Date of last access: June 30, 2014.
- [16] Edibon, Web page at <http://www.edibon.com/>, Date of last access: June 30, 2014.
- [17] Elektron, Web page at <http://www.tiendaelektron.com/>, Date of last access: June 30, 2014.
- [18] Lego Mindstorms, Web page at <http://mindstorms.lego.com/en-us/default.aspx>, Date of last access: June 30, 2014.
- [19] L. E. R. Kit, Web page at [http://www.legoeducation.us/eng/product/renewable\\_energy\\_add\\_on\\_set/2101](http://www.legoeducation.us/eng/product/renewable_energy_add_on_set/2101), Date of last access: June 30, 2014.
- [20] B. Taraghi, M. Ebner, and S. Schaffert, "Personal learning environments for higher education: A mashup based widget concept," in Proc. of 4th European Conference on Technology Enhanced Learning, Nice, France, 2009.
- [21] M. Ebner and B. Taraghi, "Personal learning environment for higher education – a first prototype," in Proc. of World Conference on Educational Multimedia, Hypermedia Telecommunications, Chesapeake, USA, 2010.
- [22] M. Palmér, S. Sire, E. Bogdanov, D. Gillet, and F. Wild, "Mapping web personal learning environments," in Proc. of Workshop on Mash-Up Personal Learning Environments (MUPPLE), held jointly with 4th European Conference on Technology Enhanced Learning (EC-TEL), Nice, France, 2009.
- [23] M. Al-Zoube, "E-learning on the cloud," Intl. Arab Journal of e-Technology, vol. 1, no. 2, pp. 58–64, 2009.

- [24] O. Casquero, J. Portillo, R. Ovelar, J. Romo, and M. Benito, "igoogle and gadgets as a platform for integrating institutional and external services," in Proc. of Workshop on Mash-Up Personal Learning Environments (MUPPLE), held jointly with 4th European Conference on Technology Enhanced Learning (EC-TEL), Maastricht, The Netherlands, 2008.
- [25] S. MacNeill and W. Kraan, "Distributed learning environments: A brief paper," Available at [http://wiki.cetis.ac.uk/images/6/6c/Distributed\\_Learning.pdf](http://wiki.cetis.ac.uk/images/6/6c/Distributed_Learning.pdf), Centre for Educational Technology and Interoperability Standards (JISC CETIS), Tech. Rep., 2009.
- [26] Lego Education: Renewable Kit, Web page at [http://www.legoeducation.us/eng/product/renewable energy add on set/2101](http://www.legoeducation.us/eng/product/renewable%20energy%20add%20on/set/2101), Date of last access: June 30, 2014.
- [27] OpenSocial, Web page at <http://opensocial.org/>, Date of last access: June 30, 2014.
- [28] Apache Shindig, Web page at <http://shindig.apache.org>, Date of last access: June 30, 2014.
- [29] Apache Shindig users, Web page at <http://shindig.apache.org/powered-by.html>, Date of last access: June 30, 2014.
- [30] LinkedIn, Web page at <https://www.linkedin.com/>, Date of last access: June 30, 2014.
- [31] Hi5, Web page at <http://www.hi5.com/>, Date of last access: June 30, 2014.
- [32] Yahoo, Web page at <http://apps.yahoo.com/>, Date of last access: June 30, 2014.
- [33] E. Bogdanov, C. Salzmann, and D. Gillet, "Widget-based approach for remote control labs," in Proc. of the 9th IFAC Symposium on Advances in Control Education, Nizhny Novgorod, Russia, 2012.

## AUTHORS

**A. C. Caminero, A. Robles-Gómez, S. Ros, Ll. Tobarra, R. Hernandez, and R. Pastor** are with the Dept. of Communications and Control Systems, Spanish University for Distance Education. ETSI Informática. C/ Juan del Rosal 16, 28040, Madrid. Emails: {accaminero, arobles, sros, llanos, Roberto, rpastor}@scc.uned.es

**M. Castro** is with the Dept. of Electrical, Electronical and Control Engineering, Spanish University for Distance Education. ETSI Industriales. C/ Juan del Rosal 12, 28040, Madrid. Email: mcastro@ieec.uned.es.

Authors would like to acknowledge the support of the following European Union projects: RIPLECS (517836-LLP-1-2011-1-ES-ERASMUS-ESMO), PAC (517742-LLP-1-2011-1-BG-ERASMUS-ECUE), EMTM (2011-1-PL1-LEO05-19883), and MUREE (530332-TEMPUS-1-2012-1-JO-TEMPUS-JPCR). Furthermore, we also thank the Community of Madrid for the support of E-Madrid Network of Excellence (S2009/TIC-1650). This article is an extended and modified version of a paper presented at the EDUCON2014 conference held at the Military Museum and Cultural Center, Harbiye, Istanbul, Turkey, 3-5 April 2014. Submitted 30 June 2014. Published as resubmitted by the authors 25 October 2014.