

Using UML Models to Describe the VISIR System

<http://dx.doi.org/10.3991/ijoe.v12i06.5707>

Razwan Mohmed Salah¹, Gustavo R. Alves², Pedro Guerreiro¹ and Ingvar Gustavsson³

¹ University of Algarve, Faro, Portugal

² Polytechnic of Porto, Porto, Portugal

³ Blekinge Institute of Technology, Karlskrona, Sweden

Abstract—Analysis and design are important phases in the system development lifecycle. In these phases, developers record information necessary for properly understanding the nature of the systems under consideration.

We consider a particular type of system, named Remote Lab that allows both teachers and students to perform real experiments over the internet and we focus on Virtual Instrument Systems in Reality (VISIR).

We describe a summarized web interface of VISIR using the Unified Modeling Language (UML). Our case study aims at identifying the requirements of VISIR and at creating a set of UML diagrams that succinctly provide enough information to both developers and users so that they acquire enough information for building a general understanding of it.

Additionally, we provide elements for extending UML to better support further user interface development in VISIR.

Index Terms—UML, RLs, VISIR, Web Pages, Activity and Use Case Diagram.

I. INTRODUCTION

In 1997, Booch, Raumbaugh and Jacobson submitted to the Object Management Group (OMG) the first version of a new modeling language, called UML, for “*Unified Modeling Language*”. Since 2001, OMG has been developing new versions of UML “graphical modeling languages” for systems engineering [1] [2]. In addition, the International Council on Systems Engineering (INCOSE) decided to make UML a standard language for system engineering [3]. It has become a de facto common language for developing and exchanging engineering designs. In addition, several organizations around the world have been using UML as a basis for design and implementation [4], for example NASA [5].

A number of UML-compatible modeling tools, such as Microsoft Visio¹ and Rational Rose², have been developed for use within academia and industry. These tools are based on functional testing concepts and input languages, both graphical and textual in nature [4]. In practice, UML is a graphical language for specifying, displaying, building, and documenting software systems. It is also a graphical notation for drawing diagrams of software

concepts. The diagrams are classified into three types: Static model, Dynamic model, and Physical model [6].

The aim of the paper is to identify the interactions occurring between users and a remote lab named VISIR system and diagram them by UML, in order to facilitate the conceptual understanding of the system structure and functionality. The tools, which are used to describe VISIR system, are focused on two kinds of models: *use case diagram* and *activity diagram*. These diagrams provide a clear idea of the workflow associated with VISIR system. In other words, the approach of the paper is to use UML and UML modeling software tools to aid understand a VISIR system by graphically describing its web interfaces.

The paper is structured as follows. Section II gives a brief overview of UML. Section III outlines remote labs, in general. An overview of features, architecture, list requirements, pages hierarchy, and present UML models “activity and use case diagrams” of the VISIR web interface is presented in section IV. Section V presents the results and suggestions of future work. Section VI concludes the paper.

II. BRIEF OVERVIEW OF UML

UML tools can be used to generate code in various languages from diagram descriptions. UML has a direct relation with object-oriented analysis and design and became an OMG standard [7]. UML includes a graphical notation that may be used to create an abstract model of a system [8]. The general purpose of UML is to make the system simple to understand and to use for users and developers [7].

In general, there are three kinds of modeling in UML [9].

- *Static Model*: includes Class diagrams, Object diagrams, and Use Case diagrams.
- *Dynamic Model*: includes Sequence diagrams, Communication diagrams, Stat Chart diagrams, and Activity diagrams.
- *Physical Model*: includes Component diagrams and Deployment diagrams.

This study focuses on two kinds of diagrams. The first one is called “Activity diagrams”, which describe the system workflow. The second one is called “Use Case diagrams”, which provide a comprehensive summary of a user’s interaction with the system in a single illustration.

¹ http://www.microsoftstore.com/store/msusa/en_US/list/categoryID.69407600

² <http://www-03.ibm.com/software/products/en/rosemod>

Activity diagrams describe a sequence of actions, starting at an initial state and ending at a final state. The initial state draws as a solid circle and it connects to the first action in the activity's sequence. An action is represented by a rectangular object with semicircular left and right ends [10].

The Use Case diagram is a scenario for understanding the interaction between users and system functions [11], rather than describing how the system works. It can be a very effective way to clarify what the system is supposed to do.

III. REMOTE LABORATORIES (RLs)

In engineering and sciences fields, laboratories are an important educational resource for students, because it's in laboratories students acquire experimental skills. Recently, several online labs have been set up to support experiments over the Internet, thus increasing the number of available experiments and the number of students able to carry them out.

The expression "online labs" refers to remote labs and virtual labs. The distinction between these two types is based on the nature of the experiments: virtual labs represent software simulated experiments whereas remote labs imply experiments with real objects and real equipment [12].

In a hands-on laboratory, students interact directly with the equipment (e.g. manipulating with the hands, pressing buttons, and so on) and receive their feedback and results directly in visual or audio ways [12] [13]. Nevertheless, there are some constraints in hands-on laboratories, especially in engineering and science fields, such as time limitation, safety, increasing number of students, and limited number of available laboratories, among others. Therefore, setting up a new laboratory at an institute or university may be expensive, in terms of the initial investment and long-term maintenance.

At present, one may find several remote laboratories serving students and teachers, as a complement (or even as a replacement) of hands-on laboratories. The interaction in RLs can be similar to that happening in the hands-on laboratories. The only difference is that the experiment in hands-on laboratory is performed physically, meaning, the students must be inside the laboratory, handling the real equipment, while in RLs, the students can perform the experiment from anywhere and at anytime [12].

RLs are a type of innovative technology-enhanced educational tools able to provide laboratory experiments to a large number of students, over the Internet. Additionally, RLs may comprehend one or more remote experiments, which offer a realistic experience, showing parameter influences and phenomena [13] [14] [15]. Furthermore, RLs provide a reusable, scalable and powerful framework [16] that can be used to share experiments among a number of different projects [17] [18], such as MIT iLabs, WeblabDeusto and NetLab.

Likewise, RLs provide many function requirements that help to extend them with new experiments and several features such as authentication, user management, and federation [17]. They can be integrated with Learning Management Systems (LMS) such as Moodle [18] and with social networks such as Facebook [17] to support students accessing them. The major advantages are availability of lab equipment (24/7), taking a small space, and

organizing time for using the lab by creating schedules for users [19].

In the general case, the architecture of a remote laboratory is built on a client-server structure [20] [21], in some cases it is based on a software controller written in LabVIEW [22].

IV. OVERVIEW OF VISIR

The Blekinge Institute of Technology (BTH) has started the VISIR project laboratory, which provides a remote electronics laboratory, in 1999. They have collaborated with National Instruments in USA and Axiom EduTech in Sweden to disseminate it by using open source technologies with other universities and organizations [23][24].

VISIR Open Lab Platform enables students to work and conduct experiments remotely and safely. The main goal of VISIR is to create distributed laboratories for collaborating in various experiments among universities via the Internet [23] [25]. Additionally, it increases the access to physical experiments for students, without a significant cost [25]. VISIR is integrated into other environments, for example MOOC, Moodle and Facebook, and it has been made available through Remote Laboratory Management Systems (RLMS) like MIT iLabs and WeblabDeusto [17] [24].

Several papers [17] [18] [24] [25] [26] [27] have been described the VISIR architecture, and clarified the overall operation cycle of VISIR. In general, they divide the VISIR system into four different independent components and then explain them in detail: Equipment server, Measurement server, Flash client, and Open labs Web layer.

Fig. 1 presents a simple navigation structure of VISIR, composed by three levels. The top level corresponds to the web interface, which allows users to interact with the software running on the web server. Between the web interface and the web server there is an application programming interface (API), which provide a set of routines, protocols and tools that are used for building the VISIR software application. This API specifies how software components should interact and be used for programming the Graphical User Interface (GUI) components. Through this API, the web site manager can easily access the VISIR system and change parameters or its functionality [28].

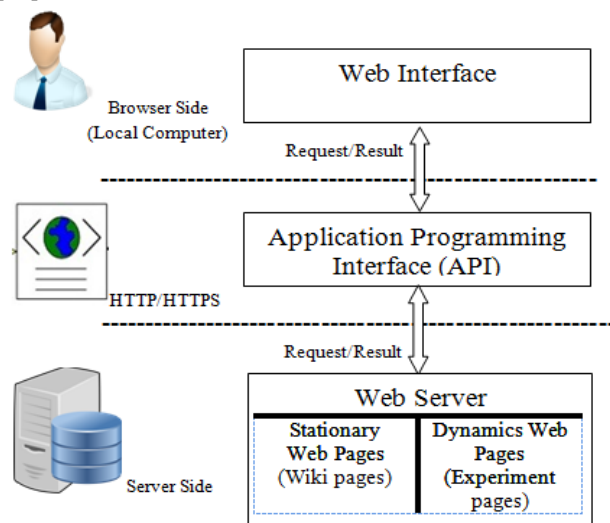


Figure 1. Navigation structure of VISIR system

The Web Server of VISIR includes stationary and dynamic pages. Stationary web pages, or Wiki pages, are static pages written in HTML. The front pages of VISIR are described through in this way¹. It should be noted that VISIR uses a server program, namely Wiki. This program allows admin to forming the content of the VISIR front pages. It provides a simplified interface that can be used for reviewing the Web page and for previewing it before publishing. Dynamic web pages, or Experiment pages, are web pages whose construction is controlled by application server processing server-side scripts.

The initial Experiment pages of VISIR were written in Flash and many VISIR laboratory nodes still use it [24] [29]. Currently, the Experiment pages have been rewritten in HTML5. Although the layout and functionality of the VISIR Experiment pages, written in Flash or HTML5, are basically the same (see Fig. 2), the use of HTML5 introduced some advantages, which will be detailed at the end of this section.

V. THE VISIR WEB INTERFACE

A. Stationary Web Pages

This subsection presents the HTML pages that form the User Interface (UI) of VISIR, excluding the part that corresponds to the actual experiments.

The VISIR entry page contains a main menu, welcome message, login hyperlink and links to pages in other languages (currently in English, Swedish, Greek, Portuguese, and Spanish). The local administrator of VISIR can add the VISIR entry page (pages at the user interface pages). For instance, the VISIR node at Porto (Portugal) exhibits an entry page different from the one exhibited by the first VISIR node, installed at Karlskrona (Sweden).

Users can login into VISIR by entering an email address and a password. After authorization, they can set up or carry out an experiment, freely and remotely.

In the main menu, the start page is a welcome user page that presents a short paragraph about nature of electronics laboratories. The history, description, manuals and references related to VISIR have been included in the about page. A user who does not have access to the system can still enter VISIR from the demo page, as a “guest”. Guests only have to enter their email and then they can run experiments available in the guest course. Some videos about VISIR are included in the demonstration part. Frequently asked questions about VISIR are included in the FAQ page, as illustrated in Fig. 3.

There are some variations in described pages from one university to another for example, the experiment pages that implemented to VISIR guests. Because VISIR is an open source project, each university may adapt it according to its requirements, for example, VISIR at BTH or VISIR at UNED [18].

B. Interaction Requirements Analysis

We now elaborate on the interactions requirements analysis of the VISIR system. The analysis process should identify all the interactions requirements, exactly. VISIR requirements have been established to satisfy the needs of the end users and to avoid errors that might occur in the system. The requirements for the analysis were collected during discussions with an administrator of VISIR at one

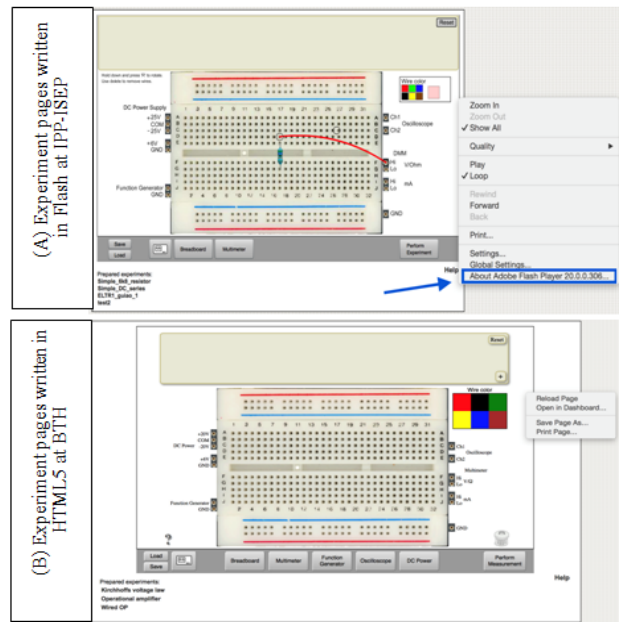


Figure 2. Experiment pages at IPP-ISEP and BTH

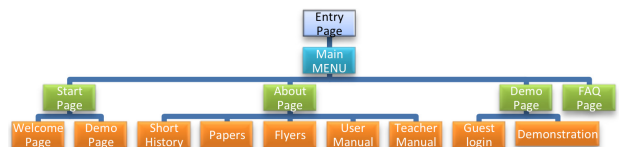


Figure 3. Hierarchy of the HTML pages forming the VISIR Stationary web pages

institution hosting it, as shown, and are listed in Table I. Afterwards; in section V we describe these requirements by using UML diagrams.

C. VISIR Experiment pages Design by UML Models

As mentioned before, this study describes VISIR by using two types of UML diagrams. First, Activity diagrams are used to describe the dynamic model associated with VISIR. Second, Use Case diagrams are used to describe the static model. The difference between static and dynamic models is that a static model shows the structural characteristics of the system, while a dynamic model shows the behavioral characteristics of the system [9] [30]. In this study, these two types of diagrams are used to describe VISIR experiment pages in detail in the two following subsections.

1) Activity Diagram for the VISIR Experiment pages

Our task now is to describe the VISIR system using activity diagrams. Users that have already registered can login to the system. Users without an active account must get one first. After that, they can login to VISIR by using their access credentials and select the experiments that are available to them. Activities performed by administrators, teacher and students are not the same. Our criteria in activity diagram focuses on the users who have credentials to access VISIR. The activity diagram describes the sequence of actions to be performed by either an administrator, a teacher or a student, from login to the preparation of a course or of an experiment or to the carrying out of an experiment. The activities selected depend on the user being an administrator, a teacher or a student. All these activities are detailed below and diagrammed in Fig. 4:

¹ <https://physicslabfarm.isep.ipp.pt/>

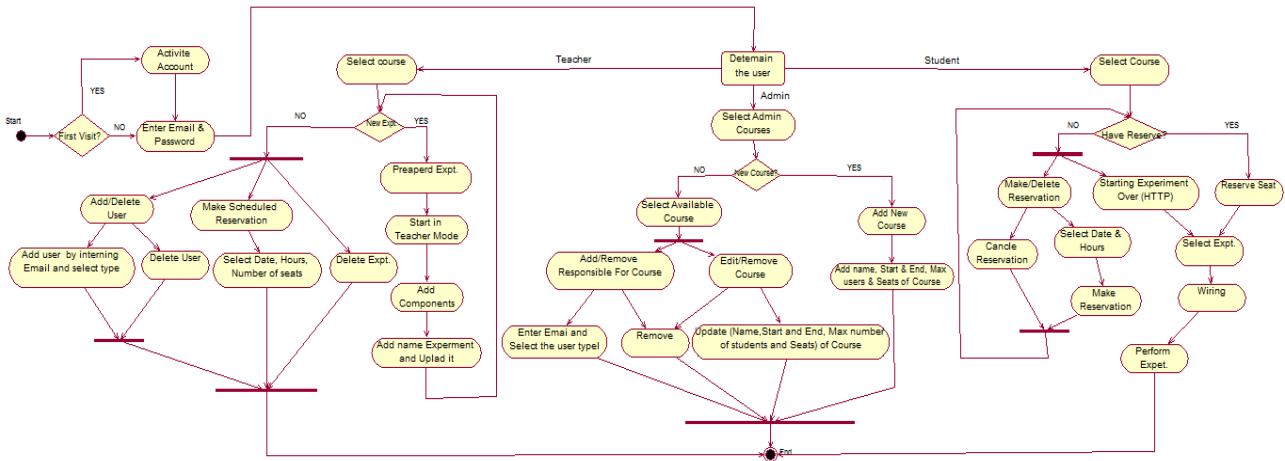


Figure 4. Activity diagram of VISIR System (Admin, Teacher, and Student)

Admin activities: after login and clicking on the admin courses, there are two activities available to the administrator. If the administrator wants to create a new course, he should click on “add course” and then fill in the fields (name of course, start and end dates, and maximum number of students and seats) in the forum. To edit the course or add the responsible of the course, he should select the available course and then change the contents fields or add the responsible. Course cancelation and changing the responsible of course are also available.

Teacher activities: after selecting the course, the teacher can perform several activities. If the teacher wants to add a new experiment in his course, he should select the add prepare experiment and then click on “Start client in teacher mode” to add the components needed for the new experiment. Afterwards, the teacher should give the name for the new experiment and upload the experiment file. After preparing an experiment, the teacher can schedule reservations by selecting the date, number of hours and seats and add new users by inserting their email in the specific field and determining their type. Deleting experiments and users is also available to the teacher as well.

Student activities: to perform experiments, students should select the course prepared by the teacher. If the student did not make a seat reservation, there are two ways to perform the experiment. The first one, without a pre-booking, is to run their experiment over HTTP and then select the experiment for a limited time. Another way is to make a reservation by selecting the date and hours. Once the ongoing reservation comes, students can start performing the experiment according to the date of the reservation. Student can also cancel the reservation.

2) Use Case Diagram for the VISIR Experiment pages

Another way to describe the VISIR system is by using static model, namely the Use Case diagram. This type of diagram is used to specify requirements, functionality and environment (how the environment should interact with the subject) [31]. The elements and relationships are shown next [32], see table II.

In this subsection, we diagrammed all the functionalities of partner users who have tasks to perform in VISIR. Before preparing new experiments and including them in VISIR, VISIR instructor must ratify those experiments. As a consequence, some experiments may need to be modified, for adaption to VISIR.




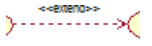
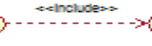
In the static model, besides admin, teacher, and student, we added two additional actors, namely technician and guest. The technician is responsible for the preparation of experiments by adding the equipment and components into its Relay Switch Matrix, adding max list, library, and circuit files into the VISIR server.

Others actors (Guest, Admin, Teacher, Student) use the VISIR system from the UI. The functionalities of those actors have been detailed and described as diagrams, see Fig. 5.

TABLE I.
 LIST OF VISIR INTREACTIONS REQUIREMENTS

REQ_ID	DESCRIPTION	BRIEF DESCRIPTION
REQ_01	Users Login	Student/ Teacher/Admin can login to the system to perform their experiment
REQ_02	Activate Account	Student/ Teacher/Admin can activate their account
REQ_03	Guests Login	Guest can login to the system to perform experiment
REQ_04	Select Course	Admin/Teacher/Student can select available course
REQ_05	Add New Course	Admin can add a new course
REQ_06	Edit/Remove Course	Admin can edit or remove course
REQ_07	Add/Remove Responsible of Course	Admin can add or remove the responsible of a course
REQ_08	Edit Wiki Pages	Admin can edit the system wiki pages
REQ_09	Edit the User Type	Admin can change the user type
REQ_10	Select Experiment	Students can select experiment
REQ_11	Add/Remove Experiment	Teacher can add or remove a prepared experiment
REQ_12	Make/ Delete Schedule Reservation	Teacher/Student can make or delete schedule reservation
REQ_13	Add/Delete Users	Teacher can invite and delete users to use the system
REQ_14	Ongoing Reservation	Students can make ongoing reservation in the VISIR system.
REQ_15	Start Experiment	Students can start experiment with reservation or without reservation
REQ_16	Reserve Seats	Students and teachers can reserve seats

TABLE II.
 USE CASE ELEMENTS AND RELATIONSHIPS

Elements	Describe and Function	Notation
Actor	Behavioral classifier that specifies a role played by an external entity that interacts with the subject (e.g. exchange, add).	
Use Case	Describe functionality that is provided by the system and determines its the requirements. It shows as an ellipse containing the name of the use case.	
Association	The communication path between an actor and a use cases that is part of the system	
Extend	The insertion of additional behavior into a base use case that does not know about it.	
Include	The insertion of additional behavior into a base use case that explicitly describes the insertion.	

Guest Use Cases: VISIR allows students who have authorization to access the experiments that are available to them. This does not mean that other users, called as guests, cannot access to some experiments in the VISIR system. In fact, VISIR has specific experiments that are available in the guest page. This facility is available to the guest user in the VISIR system. Fig. 6 shows the use cases of guest that include some specific functions. To access the experiments available to him, the guest must login by entering his email. After selecting the guest course and then clicking on the start experiment "hyperlink" over HTTP, the guest can select one of the experiments and run it freely. Finally, the guest can stop the experiments by logging out. Notice that some VISIR nodes have limited time for guests, for example the WebLab-Deusto, at the University of Deusto in Spain.

Admin Use Cases: the main functionality of the administrator in VISIR is to add a course for the users. To add a new course, he should add the course details such as course name, start and end course, add the maximum number of students, and booking seats. To edit a course, the options are update or remove course. In case of update, the admin should change the contents of the fields and then click on the update button. For removing the course, the admin only has to press on the remove button.

Moreover, he can add the course responsible by adding his email and specify the type (as instructor or teacher). Additionally, the admin can edit the VISIR wiki pages.

One of the important cases for admin is to change the user type. This means that the admin has permit to change the users' types from one case to another, for instance, teacher to students or admin and vice-versa. Moreover, there are several functionalities related to admin such as enter password manually, activate and reactivate user, and force user deletion. All admin functionalities are shown in Fig. 7.

Teacher Use Cases: after the teacher account is created by the VISIR admin, the teacher has several functionalities available in VISIR. Starting with the preparation of a new experiment, the teacher should select the course name that was created by admin. After that, he should add the components needed, name the new experiment, and upload the experiment file. Delete experiment is also available to the teacher by clicking on the cross notation.

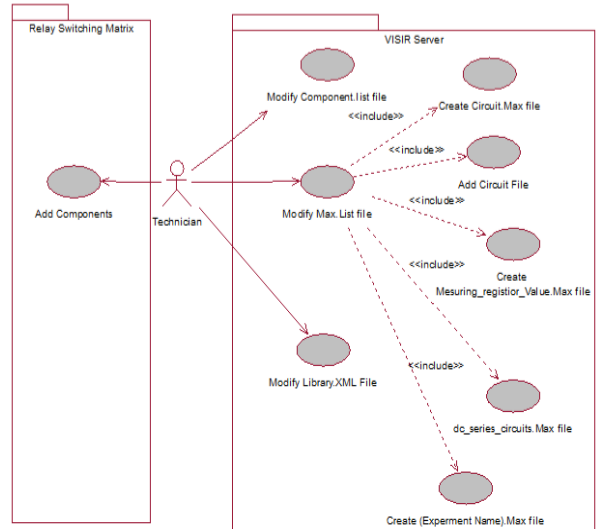


Figure 5. Technician Use Case Diagram

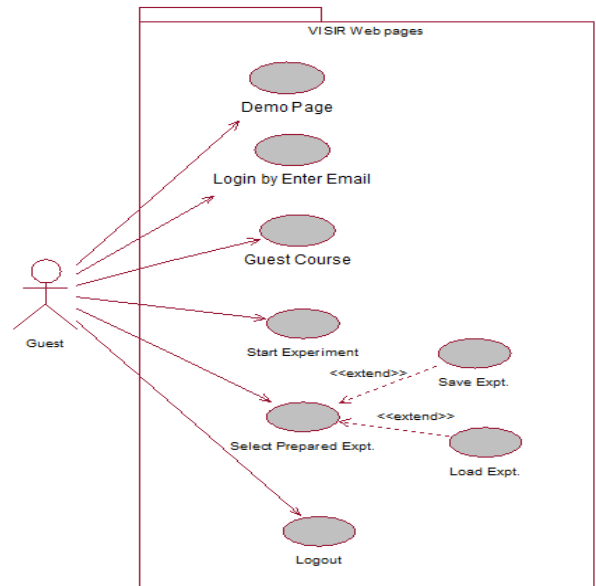


Figure 6. Guest Use Case Diagram

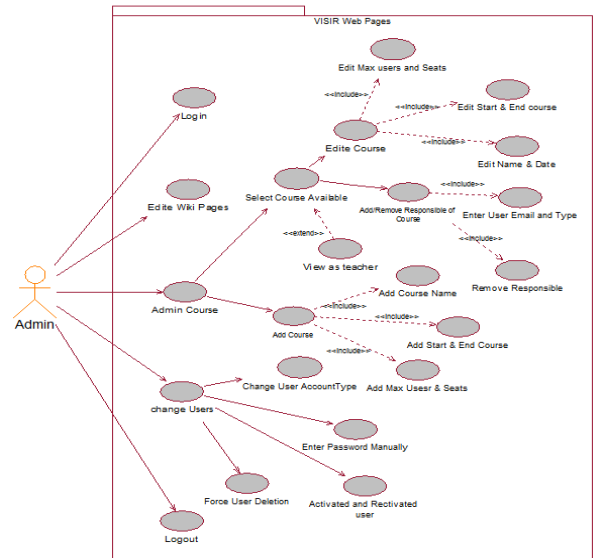


Figure 7. Admin Use Case Diagram

After the experiment has been prepared, the teacher can make scheduled reservations, which become available for the students to run the experiments. In addition, the teacher should select the date, hours, and number of seats for the experiment. In another task, he can invite more users as teachers, or as students, and add or delete them. Moreover, the teacher is awarded the access rights of a student as well. All teacher functionalities are illustrated in Fig. 8.

Student Use Cases: VISIR allows the students to perform several functions as well. After login, the students can only see the courses in which they are enrolled in and the experiments that were created by teacher. If the student has not made a reservation for the experiment in advance, he can run the experiment over HTTP. If the student has an ongoing reservation, he can start to perform the experiment directly. There are other options for students, for example save the experiment and load it again in another time. To make a new reservation, the student should select the date and hours for his experiment and it will appear in the "My reservations" list. To complete a reservation, students should also reserve the seat from teacher-scheduled reservations. Moreover, cancellation of a reservation is allowed to students. The complete functionalities are illustrated in Fig. 9.

As previously said, VISIR experiment pages have been translated from Flash to a new generation of Hyper Text Markup Language (HTML), namely HTML5. Several new features are included in HTML5 language, for instance support multimedia and different web browsers (Internet Explorer, Google Chrome, etc). In addition, it runs on low-powered devices, such as smartphones and tablets. A major difference is that HTML5 eliminates the need for the Flash player plug-in, which is facing limited support in some devices like android platform. HTML5 makes developing new web applications simpler and easier because it includes many useful markup tags and it covers web needs such as interactive and media rich, by introducing ways to embed videos, audio, and draw simple graphics [33].

For instance, it is easier to develop experiment pages in HTML5 that show several instruments at the same time, while also allowing to include additional information such as short tutorials or graphs. This is the case of a particular experiment page developed for a trial with secondary school students as described in [34, pp71]. In this customized experiment page¹ there are two digital multimeters (DMM), a DC Power Supply, and a virtual breadboard. Description of the experiment, instruments, and measurements can be in one page, rather than in the main page only². In addition, this example is just a special case with only one page, viz, it is a special case for secondary school students, as shown in Fig. 10 (a and b).

VI. RESULTS AND SUGGESTIONS

In general, VISIR has proved to be an effective educational tool, regularly serving thousands of university students around the world [35] [36]. In addition, it is also used by students from secondary schools [34], helping them to complete their experiments remotely [37].

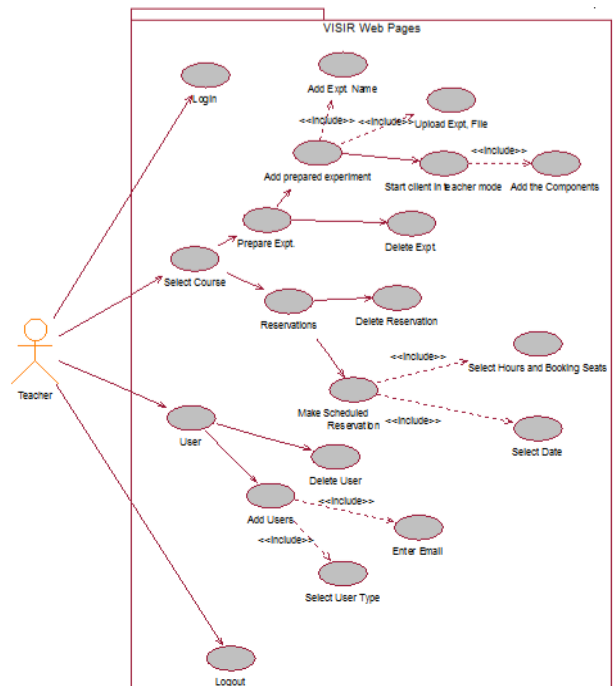


Figure 8. Teacher Use Case Diagram

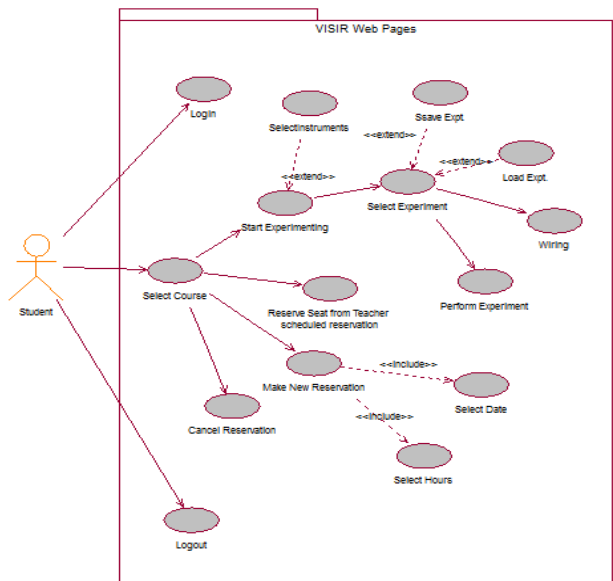
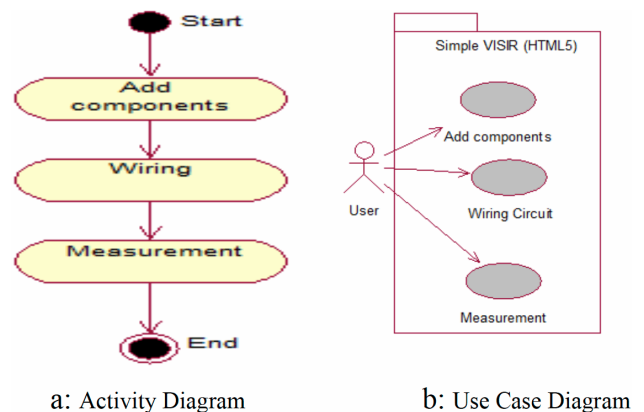


Figure 9. Student Use Case Diagram



a: Activity Diagram

b: Use Case Diagram

Figure 10. Experiment Page Example (Ohm's law) for Secondary School

¹ http://dev.openlabs.bth.se/~zeta/dav/git/test_exercise/test-instruction.html

Web interfaces of VISIR system are designed well. The main point is that the VISIR web interfaces makes it easier to run each experiment, by taking advantage of its multiple page approach. It must be mentioned that the initial VISIR experiment pages were written in Flash, and that many VISIR laboratories are still using it. Given that today the VISIR experiment pages have been rewritten in HTML5 and that current work focus on this approach, we expect a broader adoption on VISIR experiment pages with HTML5 in the near future [38].

The main facility in the HTML5 VISIR experiment pages is that it is possible and easy to add more instruments, for instance a second DMM.

VISIR System includes students (University and Secondary School), teacher, administrator, instructor, technician, and developer. They work together in the loop to improve VISIR and showing the services. Outline of diagrams, the main duty of VISIR system users is shown in Fig. 11.

A welcome consequence of this increased friendliness of VISIR, obtained by using HTML5, is that it can be used with less sophisticated users, such as secondary school students. In any case, a friendlier system means that there will be opportunity for more experiments, and this will drive up the technical skills of the students, and will help them to reach the learning objectives of their courses [24][34].

VII. CONCLUSIONS

Nowadays, UML is widely used as a visual modeling language for software systems. It is a representation of a model of a system under design, implementation, or already in existence. In addition, it has succeeded to bring the benefits of a standardized modeling syntax to business governance, allowing all aspects of all businesses to benefit from the precision and ease-of-understanding offered by a good modeling language. Moreover, it is accepted by the OMG as the standard for modeling object-oriented programs.

Recently, many universities have been using VISIR in their education system, especially the Flash implementation. In addition, VISIR has proven to be an effective technology-enhanced educational tool. The features of VISIR such as accessibility, adaptation and open-source code led to its inclusion in and adaptation to some online labs projects.

In this paper, we focused on UML-based visual modeling to specify the VISIR requirements and interactions. We presented UML models for the VISIR web interface pages. These models have two components: a dynamic model (represented by Activity diagrams) and a static model (represented by Use Case diagrams) to describe the VISIR actions.

We believe that those UML models can help the VISIR community to better understand the flow process associated with the VISIR system. These included diagrams that were prepared having in mind developers, users, and anybody interested in understanding the VISIR system functionalities.

In conclusion, RVLs technologies have proven to be effective and useful tools for students and teachers to develop their skills. The purpose of this study is to facilitate the understanding of how VISIR works, by using UML dia-

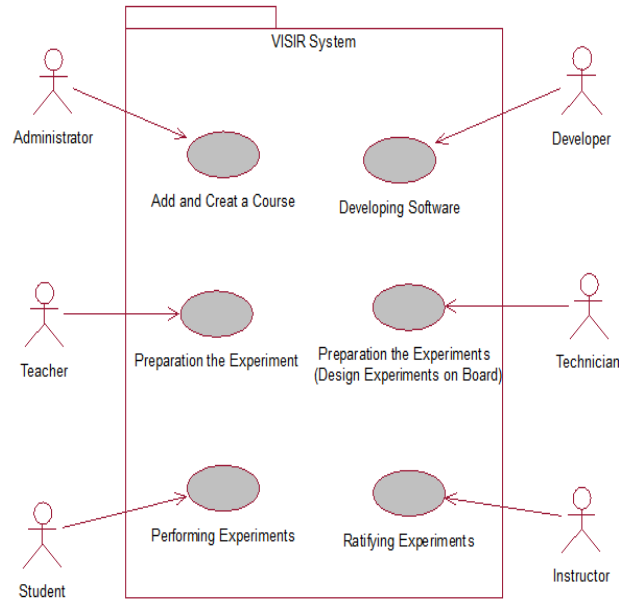


Figure 11. Duties Users of VISIR System

grams, and to highlight the characteristics of VISIR interfaces currently in use. The new language, i.e. HTML5, has helped to create applications in a simple and faster way without any plug-ins. As a result, it will be a good idea to extend the original RVLs, wrapping them in easier and simpler interfaces, suitable for secondary school students, for example. In this way, we believe that in the near future, VISIR will be widely available to students in all study levels.

ACKNOWLEDGMENT

We thank Johan Zackrisson at Blekinge Institute of Technology and Frederico L. Jacob at Oporto Polytechnic Institute- ISEP for assistance with technical part, and comments that greatly improved the manuscript.

REFERENCES

- [1] M. Akerholm, I. Crnkovic, G. Mustapic and M. Davidsson, "Introduction for using UML", Edition2, 2006.
- [2] P. Mihnev and T. Zafirova-Malcheva, "A general model for educational software design and development," In 7th International Conference on Education and New Learning Technologies, 2015, Barcelona, Spain, 2015, pp. 4079-4089.
- [3] T. Weikiens, "Systems engineering with SysML/UML: modeling, analysis, design", Morgan Kaufmann, Burlington, MA, 2011.
- [4] J. Hartmann, C. Imoberdorf, and M. Meisenger, "UML-Based integration testing," In ISSTA 2000 conference proceedings, Portland, Oregon, 22-25 August 2000, pp. 60-70.
- [5] Watson, "Visual modeling: past, present and future," Online, Object Management Group, http://www.uml.org/Visual_Modeling.pdf, December 2014.
- [6] R.C. Martin, "UML for java programmers," Prentice Hall Professional Technical Reference, 2003.
- [7] Tutorials point, "UML overview," [Online] <http://www.uml-diagrams.org/use-case-diagrams.html>, December 2014.
- [8] MediaGroup1 LLC, "Unified modeling language planning with diagrams," [Online] <http://www.dreamincode.net/forums/topic/49062-unified-modeling-language/>, December 2014.
- [9] Software Development Videos and Tutorials, "Static and dynamic UML models or diagrams," [Online] <http://www.softdevtube.com/2010/11/11/static-and-dynamic-uml-models-or-diagrams/>, December 2014.

PAPER
USING UML MODELS TO DESCRIBE THE VISIR SYSTEM

- [10] D. Bell, "UML basics: Part II: The activity diagram. IBM 2003," [Online] https://www.ibm.com/developerworks/rational/library/content/RationalEdge/sep03/f_umlbasics_db.pdf, December 2014.
- [11] F.T. Sheldon, K. Jerath, Y. Kwon, and Y. Baik, "Case Study: implementing a web based auction system using UML and component-based programming", In *Computer Software and Applications Conference*, 2002. COMPSAC 2002. Proceedings. 26th Annual International, 2002, pp. 211-216.
- [12] J. R. Brinson, "Learning Outcome Achievement In Non-Traditional (Virtual, Remote) Versus Traditional (Hands-On) Laboratories: A Review of the Empirical Research," In *Computers & Education*, Vol 87, pp. 218-237, 2015.
- [13] S. Frerich and M. Petermann, "Virtual labs and remote labs: practical experience for everyone," In *Global Engineering Education Conference (EDUCON)*, 2014 IEEE, 2014, pp. 312- 314.
- [14] Remote laboratories, "What are remote laboratories?," [Online], <http://remotelaboratory.com/remotelaboratories/what-are-remotelaboratories/>, December 2014.
- [15] J. Garcia-Zubia and G.R. Alves, "Using Remote Labs in Education: Two Little Ducks in Remote Experimentation (Eds.)," University of Deusto Press, Bilbao, 2011. ISBN 978-84-9830-335-3.
- [16] J. Faias, J. M. Ferreira, and C. Leite, "Accessibility and usability of online labs: real barriers in a virtual world," in *Remote Engineering & Virtual Instrumentation Conference*, 2007, pp.1-6.
- [17] L. Rodriguez-Gil, P. Orduna, J. Garcia-Zubia, and D. Lopez-de-Ipina, "Advanced integration of OpenLabs VISIR (Virtual Instrument Systems in Reality) with Weblab-Deusto," In *Remote Engineering & Virtual Instrumentation Conference*, 2012, pp. 1-7.
- [18] M. Tawfik, E. Sancristobal, S. Martin, R. Gil, G. Diaz, A. Colmenar, J. Peire, M. Castro, K. Nilsson, J. Zackrisson, L. Hakansson and I. Gustavsson, "Virtual instrument systems in reality (VISIR) for remote wiring and measurement of electronic circuits on breadboard," *IEEE Transactions on Learning Technologies*, vol. 6, pp. 60-72, 2013.
- [19] J. Garcia-Zubia, U. Hernandez, D. Ponta, P. Orduna, I. Trueba, and I. Angulo, "WebLab-GPIB at the University of Deusto," In *Remote Engineering & Virtual Instrumentation Conference*, 2007, pp.1-7.
- [20] L. Yaoye, K. E. Sven and C. Constantin, "A Scheduling system for shared online laboratory resources," 38th ASEE/IEEE Frontiers in Education Conference, October 22-25, New York, NY, T2B-1, 2008.
- [21] P. Orduna, J. Garcia-Zubia, J. Irurzun, E. Sancristobal, S. Martin, M. Castro, D. Lopez-de-Ipina, U. Hernandez, I. Angulo, and J. M. Gonzalez, "Designing experiment agnostic remote laboratories," In *Remote Engineering & Virtual Instrumentation Conference*, 2009.
- [22] A. Mani, R. S. S. Prasanth and C. Patvardhan, "Design of Web-Based experiments on acceleration and speed transducers," *Hindawi Publishing Corporation Education Research International* Volume 7, pp. 1-7, 2011. <http://dx.doi.org/10.1155/2011/426062>
- [23] J. Zackrisson, I. Gustavsson, and L. Hakansson, "An overview of the VISIR open source software distribution," In *Remote Engineering & Virtual Instrumentation Conference*, 2007, pp.1-5.
- [24] R.M. Salah, G.R. Alves, D. H. Abdulazeez, P. Guerreiro and I. Gustavsson, "Why VISIR? Proliferative activities and collaborative work of VISIR system," In *7th International Conference on Education and New Learning Technologies*, 2015, Barcelona, Spain, 2015, pp 3824-3835.
- [25] I. Gustavsson, J. Zackrisson, L. Hakansson, I. Claesson and T. Lago, "The VISIR project - an open source software initiative for distributed online laboratories," In *Remote Engineering & Virtual Instrumentation Conference*, 2007, pp.1-6.
- [26] E.S. Ruiz, M. Tawfik, S. Martin, R. Gil, A. Pesquera, F. Garcia, V. J. Rodriguez, F. Mur, M.J. Albert, C. Perez, G Diaz, and M. Castro, "Design, development and implementation of remote laboratories in distance electronics, control and computer subjects," In *Remote Engineering & Virtual Instrumentation Conference*, 2013, pp. 1-5.
- [27] I. Gustavsson, K. Nilsson, J. Zackrisson, L. Hakansson, J. Garcia-Zubia, G. Alves, U. Hernandez, R. Costa, T. Lago, and I. Claesson, "The VISIR Open Lab Platform 5.0 – an architecture of a federation of remote laboratories", In *Remote Engineering & Virtual Instrumentation Conference*, 2011, pp. 284-288.
- [28] RK, "Social Media ROI for Beginners," [Online] <https://sociolytics.wordpress.com/2011/07/11/social-media-platform-apis-what-why-and-how/>, 2012, February, 2014.
- [29] M. Tawfik, E. Sancristobal, S. Martin, R. Gil, P. Losada, A. Pesquera, A. Robles-Gomez, A.C. Caminero, S. Ros, R. Pastor, R. Hernandez, G. Diaz, J. Peire, and M. Castro, "Remote Experimentation in the Electronic Engineering Field: VISIR@UNED," In presented at the VII International Conference on Engineering and Computer Education - ICECE '2011 (Workshop), Guimaraes, cradle of Portugal, 2011, pp. 39-47.
- [30] NIIT, "Understanding dynamic modeling" [Online] http://www.niitstudent.com/india/Content/063OUML1S1/OEBPS/13_ch07.htm, December 2014.
- [31] UML diagrams, "UML use case diagrams", [Online]
- [32] <http://www.uml-diagrams.org/use-case-diagrams.html>, December 2014.
- [33] J. Rumbaugh, I. Jacobson, and G. Booch, "The unified modeling language reference manual," Addison-Wesley Longman Ltd., Essex, UK, 1999.
- [34] S. Basu, "10 Websites to See What HTML5 Is All About," 2010, [Online], <http://www.makeuseof.com/tag/10-websites-html5/>, July, 2014.
- [35] L. Claesson, "Remote Electronic and Acoustic Laboratories in Upper Secondary Schools" [Electronic resource (Thesis)], 2014. - ISBN: 978-91-7295-284-3.
- [36] A. V. Fidalgo, M. C. Costa Lobo, M. A. Marques, M. C. Viegas, G. R. Alves, J. Garcia-Zubia, U. Hernández, and I. Gustavsson, "Using remote labs to serve different teacher's needs A case study with VISIR and RemotElectLab," In *Remote Engineering and Virtual Instrumentation (REV)*, 2012 9th International Conference on, 2012, pp. 1-6.
- [37] G. Diaz, F. Garcia Loro, M. Castro, M. Tawfik, E. Sancristobal, and S. Montoso, "Remote electronics lab within a MOOC: Design and preliminary results," In *Experiment@ International Conference (exp.at'13)*, 2013 2nd, 2013, pp. 89-93.
- [38] R.M. Salah, G.R. Alves, B. Datkiewicz, P. Guerreiro and I. Gustavsson, "VISIR System @ BTH, DEUSTO, ISEP, AND UNED Institutes: Assisting and Supporting Hands-on Laboratories to Serve Higher Education Students," In *8th International Conference of Education, Research and Innovation*, 2015, Seville, Spain, 2015, pp. 2535-2547.
- [39] [Expressoemprego.pt](http://expressoemprego.pt), "Português lidera expansão de laboratório remoto de topo," [Online] <http://expressoemprego.pt/noticias/portugues-lidera-expansao-de-laboratorio-remoto-de-topo/3949>, January, 2016. (In Portuguese)

AUTHORS

Razwan Mohmed Salah received his BSc degree in computer science from the University of Technology (UoT), Baghdad-Iraq, and MSc degree in information technology (IT) from the University Utara Malaysia (UUM), Kedah-Malaysia. Currently, he is pursuing PhD in Online Labs fields at Faculty of Science and Technology, Universidade do Algarve (UAlg), Faro-Portugal, and partially at Polytechnic of Porto-ISEP. His research interests include remote labs, E-learning, and education technology. (E-mail: razwan.mayi@gmail.com).

Gustavo R. Alves graduated in 1991 and obtained an MSc and a PhD in Computers and Electrical Engineering in 1995 and 1999, respectively, from the University of Porto, Portugal. He is with the Polytechnic of Porto - School of Engineering, since 1994. He was involved in several national and international R&D projects and has authored or co-authored more than 160 conference and journal papers with a referee process. His research interests include engineering education, remote labs, and design for debug & test. (E-mail: gca@isep.ipp.pt).

Pedro Guerreiro graduated in Computer Science at the New University of Lisbon, in 1977. He then obtained a doctorat de troisième cycle at the University of Grenoble, in 1981, and a PhD, again at the New University of Lisbon, in 1983. From then on he pursued a teaching and research career at the New University of Lisbon, until he moved, in 2007, to the University of Algarve, where he is now a full professor of Computer Science. His research interests are Programming, Programming Languages, Algorithms, Competitive Programming, Software Engineering, Computer Science Education and e-Learning. (E-mail: pjg Guerreiro@ualg.pt).

Ingvar Gustavsson received the M.S.E.E. and Dr. Sc. degrees in electrical engineering from the Royal Institute of Technology (KTH), Stockholm, in 1967 and 1974, respectively. Together with another research scientist, he founded a private company providing automatic inspec-

tion systems for industrial customers in 1983. In 1994, he returned to the academic world to take up his current position as ASSOCIATE PROFESSOR of electronics and measurement technology at Blekinge Institute of Technology (BTH), Sweden. In 1999, he started a remote lab project at BTH that today is known as VISIR (Virtual Instrument Systems in Reality). The GOLC Online Laboratory Award in the category “Remote Controlled Lab” was presented to VISIR in 2015. Dr. Gustavsson partly retired from office 2012 making it possible to concentrate on activities related to VISIR. He is still a member of Swedish professional societies and of the IEEE. His research interests are in the areas of instrumentation, remote labs, industrial electronics, and distance learning. (E-mail: ingvar.gustavsson@bth.se).

Submitted 31 March 2016. Published as resubmitted by the authors 03 May 2016.