Towards an Online Lab Portal for Inquiry-based STEM Learning at School

Sten Govaerts[1], Yiwei Cao[2], Andrii Vozniuk[1], Adrian Holzer[1], Danilo Garbi Zutin[3], Elio San Cristóbal Ruiz[4], Lars Bollen[5], Sven Manske[6], Nils Faltin[2], Christophe Salzmann[1], Eleftheria Tsourlidaki[7], and Denis Gillet[1]

[1] EPFL, REACT, Station 9, CH-1015 Lausanne, Switzerland
{firstname.lastname}@epfl.ch
[2] IMC AG, Scheer Tower, Uni-Campus Nord, 66123 Saarbrücken, Germany
yiwei.cao@im-c.de, nils.faltin@im-c.de
[3] Center of Competence in Online Laboratories and Open Learning,
Carinthia University of Applied Sciences, Villach, Austria
d.garbizutin@fh-kaernten.at
[4] DIEEC, UNED, C. Juan del Rosal, 12, 28040, Madrid, Spain
elio@ieec.uned.es
[5] University of Twente, Department of Instructional Technology,
PO Box 217, 7500 AE Enschede, The Netherlands
l.bollen@utwente.nl
[6] COLLIDE Research Group, University of Duisburg-Essen, 47047 Duisburg, Germany
manske@collide.info
[7] Ellinogermaniki Agogi Sxoli Panagea Savva, Dimitriou Panagea str., 15351 Pallini, Greece
eleftheria@ea.gr

Abstract. Nowadays, the knowledge economy is growing rapidly. To sustain future growth, more well educated people in STEM (science, technology, engineering and mathematics) are needed. In the Go-Lab project we aim to motivate and orient students from an early age on to study STEM fields in their future educational path by applying inquiry learning using online labs. This paper presents an inquiry learning portal where teachers can discover, use and enhance online labs appropriate for their courses and students can acquire scientific methodology skills while doing experiments using the labs. The Go-Lab portal architecture is presented, which contains a repository of online labs, inquiry learning spaces and complementary services. The paper discusses a first version of the portal and our future plans.

Keywords: online lab, portal, design, inquiry learning, metadata, architecture, interoperability, lab repository

1 Introduction

In this day and age the knowledge economy [7] expands rapidly, we need to motivate more young people to choose science and engineering as their future career path to keep our future economy competitive. To engage students in science and technology,

various reports [17, 16] suggest to include scientific inquiry in their courses. The Go-Lab project[8] [11] aims to encourage students to engage in scientific topics and get acquainted with scientific inquiry methodologies through the use of remote laboratories, virtual experiments and data sets (henceforth simply referred to as 'online labs').

Online labs are a good means to teach scientific inquiry skills to students and an aid for teachers to illustrate scientific theory [11]. Inquiry learning typically leads students through various phases, e.g. orientation, conceptualization, investigation, conclusion and discussion, where students create hypotheses, evaluate them through experiments and then reflect on them, possibly repeating the cycle. This type of learning shows benefits over typical lectures or demonstrations [12].

Although labs for teaching have become available recently, a widely used online lab portal integrated with a ready-to-use learning environment is still missing (see Section 3). Usually, individual online labs are operated, maintained and promoted by the lab owners, which causes a high operational cost and limited access. Through the Go-Lab portal, we aim to establish a federation of online labs where lab owners can promote their labs, and teachers can find labs to support their activities and share their resources with others. We do not aim to replace teachers, but rather enable teachers to support and aid students who are working with online labs.

This paper presents the first two steps in the development life-cycle [1] of the portal, namely requirement analysis and systems design. Section 2 describes the requirements for the Go-Lab portal. Based on this, Section 3 compares existing portals. Afterwards, Section 4 presents the architecture of the portal and Section 5 discusses the implementation. Finally, Section 6 concludes the paper.

2 Requirements of the Go-Lab portal

This section focuses on the first step of the development life-cycle [1], namely requirements analysis, by first setting a common terminology, identifying the main portal users, describing the usage and afterwards discussing requirements.

2.1 Terminology

Online labs are remote laboratories, virtual experiments or data sets accessible from the browser through apps. *Apps* are Web applications (e.g. OpenSocial gadgets), for example to operate a lab or support learning (e.g. via scaffolding).

Inquiry learning spaces (ILS) are learning environments that can contain labs, learning resources and apps to enable inquiry learning. *Learning resources* are typically texts, videos and other materials to assist and assess students. Teachers usually set up an ILS for their students. An ILS can be shared with other teachers who can repurpose and adapt it to fit their purpose.

[8] The Go-Lab project, `http://www.golab-project.eu`

2.2 Users, user story, functional and non-functional requirements

We identify three types of portal users: *lab owners*, *teachers* and *students*. A lab owner is a user who operates and publishes a lab. A teacher is a user who teaches with an ILS. A student is a user who carries out inquiry learning activities in an ILS. The following user story provides more context on the portal use.

User story. On the Go-Lab portal John searches for interesting activities for his physics course. While browsing through the 'Big Ideas' in science section, he finds an appropriate topic about particle interactions, where he selects the 'Conservation of Momentum' activity, using the 'Hypathia' lab by CERN. He opens the activity in the learning environment to try it out. The observation phase and its learning resources are shown and he can edit the resources and apps. An available learning scenario instructs how he can teach with this activity. John decides to add a recommended scaffolding app. Additionally, he rewrites some of the explanations and tips to better fit the knowledge of his students. After tweaking each inquiry learning phase, he teaches with this learning environment in class. Afterwards, he decides to publish his environment back on the Go-Lab portal and share it on Facebook with his colleagues.

Functional requirements analysis. In this section we focus on the main functions of the portal, required by the portal users to fulfill their needs.

- *Publishing labs.* Lab owners publish a lab and describe it with metadata.
- *Creating ILS.* Lab owners create ILS to demonstrate a lab and teachers create ILS for students.
- *Modifying ILS.* Teachers adapt existing ILS, e.g. localize the content to the mother tongue of their students or adapt it for a different age group.
- *Publishing ILS.* Teachers publish their ILS to enable reuse.
- *Using ILS.* Teachers run activities using ILS. Students use ILS provided by teachers to conduct experiments.
- *Supporting Apps.* Students practice inquiry learning through apps (e.g. a hypothesis app or online lab apps). Teachers monitor student progress through learning analytics apps.
- *Supporting Learning Scenarios.* Teachers create learning scenarios for ILS. Students use learning scenarios provided by teacher when studying in ILS.
- *Searching Labs & ILS.* Teachers search for labs and ILS using various search filters, e.g. age and scientific domain.
- *User management.* Users log in only once to access all portal resources. Several login options are provided depending on the acceptable privacy level.
- *Social features.* Teachers and lab owners tag, comment and rate labs and ILS, and share them on social networks.
- *Tracking user activities.* The activities of portal users are tracked and used for learning analytics, recommendation and scaffolding apps.
- *Recommendation.* Recommendation of labs, ILS and apps are provided when searching, creating and editing ILS and labs.
- *Scaffolding.* Students receive assistance from scaffolding apps (e.g. prompts and feedback) based on learning analytics and teacher configurations.

Non-functional requirements analysis. Apart from the previous requirements, there are also non-functional requirements that impact the design.

Go-Lab needs to provide a common, ubiquitously accessible ILS platform, so schools do not need to spend resources on installing and administering software. The portal should federate labs and other lab repositories. To provide this federation and to support a common ILS platform, interoperability of the labs is essential. When targeting school students, special attention to usability and data privacy (e.g. anonymizing the tracked user activities) is needed. The Go-Lab project aims to support 1000 schools in 15 countries, which requires a high scalability and availability of the portal, as well as internationalization support.

3 State of the art

Existing portal solutions for online labs are reviewed and summarized in Table 1 and we evaluate their fit with the requirements presented above. We identified five main solutions in the research literature, namely the GOLC consortium's Lab2Go[9] portal [4], the Library of Labs (LiLa)[10] [15] portal created by a European consortium using MIT's iLab Shared Architecture (ISA)[11] [10], the LabShare[12] [6] portal initiated by an Australian consortium, and the University of Deusto's WebLab-Deusto[13] [8].

Functional requirements	Lab2Go	LiLa	ISA	LabShare	WebLab
Publishing labs	+	+	+	+	+
Creating ILS	-	∼	-	-	-
Modifying ILS	-	∼	-	-	-
Publishing ILS	-	∼	-	-	-
Using ILS	-	∼	+	+	+
Supporting Apps	-	-	-	-	-
Supporting Learning Scenarios	-	+	-	?	∼
Searching Labs & ILS	+	+	-	-	-
User management	∼	∼	+	+	+
Social features	+	+	-	∼	-
Tracking user activities	-	∼	∼	+	+
Recommendations	-	-	-	-	-
Scaffolding	-	-	-	-	-

Table 1. Fit between existing portals and Go-Lab requirements. Legend: requirement supported (+), partially supported (∼), not supported (-), unknown (?).

Table 1 shows that the main requirements which are met by the existing portals are *publishing labs*, *searching for labs & ILS*, *using ILS* and possibly *tracking user activities*. Furthermore, several of these portals offer some kind of user management system

[9] http://www.lab2go.net
[10] https://www.library-of-labs.org
[11] http://ilabcentral.org
[12] http://www.labshare.edu.au
[13] https://www.weblab.deusto.es

and provide some social features. Unfortunately, several requirements are not properly supported by any of the portals, such as facilities to create, modify and publish ILS. Furthermore, recommendation and scaffolding are both not supported by any portal. In a nutshell, existing portals mainly work as repositories of labs and provide only support to lab owners to publish labs and to teachers to find and use labs. As they do not provide their own learning environment, it is difficult for such portals to empower teachers by not supplying them with adequate support for modifying, reusing and publishing ILS. Go-Lab precisely aims to fill this gap by providing support for both lab consumption and lab creation.

4 The Go-Lab portal architecture

Based on the requirements, this section describes the Go-Lab architecture. For its design, we have applied several software design principles [20]. Foremost, the design should be *loosely coupled*, meaning that each component requires little knowledge of the definitions of the other components [20]. This principle enables abstraction of each component, which allows a design that can change over time. Additionally, each component should have *high cohesion*, which refers to the degree to which a component is semantically self-contained [20]. Applying *'separation of concerns'* enables modularity, as each component takes care of a separate task [20]. Finally, the design should support *subsetability*, which is the ability to produce subsets of the system. This allows us to follow an iterative and incremental development strategy and have a basic portal available soon.

4.1 Overall architecture

The high-level Go-Lab architecture, illustrated in Fig. 1, consists of two main components with a graphical user interface (GUI), namely the *lab repository* and the *ILS platform*. Both are supported by components for *user management* and tracking user activities for *learning analytics and recommendation*. By splitting up the portal functionality in this way, each component serves a very different purpose and we aim to satisfy the requirements and design principles stated above. The components have well-specified interfaces and protocols, which allow interchangeability (e.g. the ILS platform could use another repository that implements the same specification of the `Publisher` & `Instantiator` interface) and other third-party platforms can make use of each component separately enabling wider adaption of Go-Lab technology.

The next section elaborates on the components of the architecture.

4.2 Components and interface specification

The Lab Repository stores labs, apps and inquiry space templates (or ILS templates), together with their metadata. An ILS template describes the structure and content (i.e. the labs, apps & resources) of an ILS. Additionally, an ILS template can also contain a learning scenario provided by teachers that describes how to use the ILS in a pedagogical context.

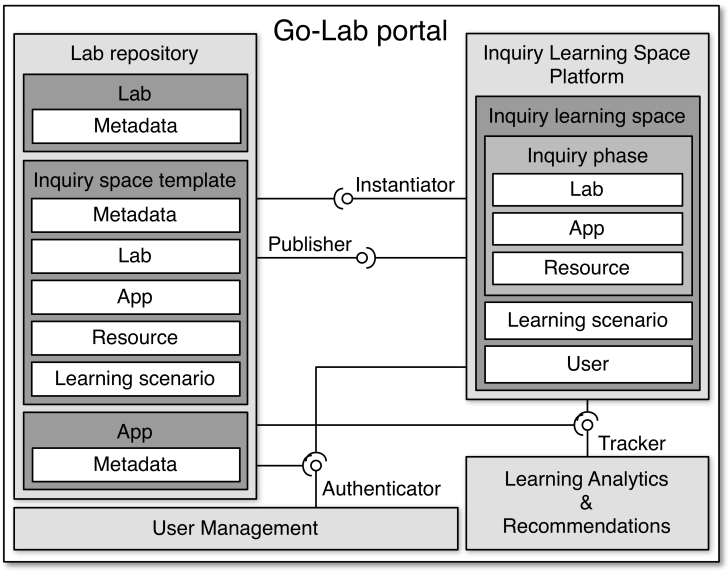


Fig. 1. The architecture of the Go-Lab Portal.

The Inquiry Learning Space Platform (ILS platform) allows teachers and students to use labs and apps in an ILS for inquiry learning. Teachers can create an ILS consisting of labs and apps available from the lab repository through the `Instantiator` interface and enrich the ILS with uploaded or online resources. Afterwards, teachers can provide students with access to the ILS, where students can conduct experiments.

Such an ILS can also be published on the lab repository via the `Publisher` interface. While publishing an ILS, teachers provide metadata that describes the ILS together with pedagogical information and possibly a learning scenario. By publishing an ILS template to the lab repository, other teachers can find it there, reuse it in the ILS platform using the `Instantiator` interface and adapt it to the needs of their students.

The *Learning Analytics and Recommendation* component collects user activities through the `Tracker` interface from the lab repository and the ILS platform that can anonymize the data for privacy reasons. The collected data is used to provide teachers with learning analytics apps to monitor student progress; lab owners can monitor the use of their labs, while students benefit through scaffolding apps. The tracked user activities are also employed for personalization of the portal, e.g. through recommendation of apps, labs and resources.

The *User management* component is in charge of user authentication and user profile management through the `Authenticator` interface to the lab repository and ILS platform enabling single sign-on for the portal.

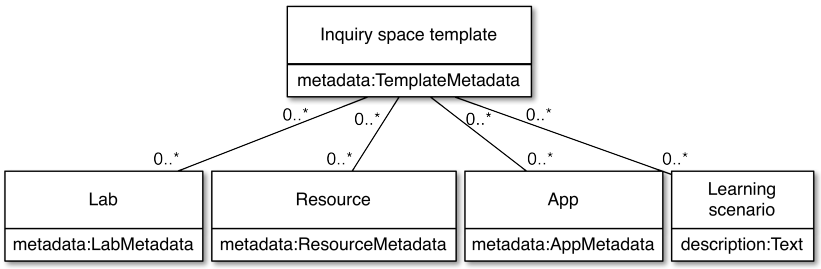


Fig. 2. Go-Lab metadata overview.

4.3 Portal interoperability

To achieve a federation of labs and to increase the potential uptake of the Go-Lab software, interchangeability of the portal components and interoperability using open specifications is important. The Go-Lab architecture achieves the technical, syntactic, semantic and pragmatic interoperability levels of the Conceptual Interoperability Model [19]. This section elaborates on interoperability enabled through the labs, metadata and the interfaces between components.

Lab interoperability enables the integration of labs with an ILS platform, which is often difficult due to the wide variety of labs and their technical differences (e.g. implemented as a Java Applet or a Flash application). To make labs interoperable with learning environments different approaches are possible. For instance, the LiLa project [15] bundles labs in SCORM [3] packages, but this does not always enable proper interoperability since SCORM is not designed for interactive labs and the support of the latest versions of SCORM by learning environments is low. In Go-Lab, online labs will be provided as *smart devices* [9] that make labs more ubiquitous, autonomous and self-aware. The smart device paradigm abstracts the details of each lab on the server-side by providing a specified set of web services [9]. This interoperability layer allows the ILS platform to run any lab supporting the smart device paradigm and smart device compatible apps can be reused to operate numerous labs.

Making existing online labs smart device compliant might require the implementation of the specified web services. In some cases it will be impossible to change the lab implementation. To enable interoperability with such labs, we will provide a *smart gateway* that transforms existing labs to be conform to the smart device specification [9]. In this way interoperability between any online lab and the ILS platform is enabled.

Metadata interoperability is the ability to exchange metadata with minimal loss of content and functionality between different systems [14]. Several initiatives [22, 18] are currently working on metadata specifications to describe online labs and related resources. Their main reason is to allow wider discovery of online labs, but metadata can provide more benefits. Apart from search and discovery of labs and apps in the lab repository, metadata is also used to exchange data between the portal components

to enable interoperability and exchangeability. For instance, this interoperability and exchangeability allows exchanging the lab repository with a third-party repository that applies the same interface and metadata specification; similarly the ILS platform could be switched. In the Go-Lab project, metadata is used to describe labs, apps, resources and ILS templates (see Fig. 2) in a linked data approach. The metadata specification will be based on a combination and extension (based on Go-Lab requirements) of the ROLE Ontology and the GOLC specification [18] and is currently being finalized. The reuse of existing open specifications will provide access to existing labs and resources, as well as services.

Interface interoperability allows different implementations of components of the Go-Lab portal to be interchanged. This can be achieved by specifying the component interfaces and the data that is exchanged (metadata interoperability). For instance, the lab repository could use another learning environment that specifies the `Instantiator` and `Publisher` interface. Additionally, the `Authenticator` interface enables the interchangeability of, for instance, the default user management with an LDAP implementation.

5 Implementation of the Go-Lab portal

Based on the presented design, we have started the portal implementation following an iterative and incremental approach. In parallel with the architectural design, we have also designed a GUI mockup[14] using participatory design.

The lab repository is implemented on top of the existing ROLE Widget Store [5] that is built with Drupal[15]. Drupal is a widely used, open source content management system that allows high scalability. Labs, apps, resources and ILS templates will be described with metadata (e.g. functionalities & 'Big Ideas' in science categories), which is used to organize and search for labs, apps and ILS templates. The existing functionality for learning scenarios in the ROLE Widget Store is reused. In a first implementation phase, we aim to implement publishing labs, apps and ILS templates and integration with the ILS platform through the `Instantiator` interface.

The ILS platform is implemented on top of the Graasp platform [2], which is a social media platform that supports personal and collaborative activities using resources and OpenSocial apps. Currently, Graasp already allows teachers to build a basic ILS. In a first phase, this functionality will be further extended and a dedicated GUI for students is under development. Part of the learning analytics and recommendation component will be integrated with Graasp using the OpenSocial specification together with the ActivityStreams specification to represent the user activity data [21].

In a later phase, the user management component and other more advanced functionality will be implemented. The full metadata schema will be gradually implemented.

[14] The mockup is available at `http://www.go-lab-project.eu/content/prototypes`

[15] Drupal, `http://www.drupal.org`

6 Conclusion & future work

The architecture described above aims to satisfy the requirements and design principles. Each of the components in the architecture handles a well-defined coherent set of tasks, which fulfils the *'separation of concerns'* design objective. Through well-defined interfaces and metadata specifications, we enable a *loosely coupled* architecture with *high cohesion*. The portal architecture follows the *subsetability* design principle and thus allows an iterative and incremental development. Moreover, this enables early deployment in real-life settings, which will be exploited for participatory design evaluations.

One of the main requirements has been achieved, i.e. to provide a common, ubiquitously accessible inquiry learning environment without any installation hassle. Various design decisions contribute to this. First, the smart device specification and smart gateway for online labs empower interoperability between any lab and the ILS platform. Second, the Go-Lab portal provides a seamless integration between a repository of labs and a learning environment. The integration is supported by the well-defined interfaces and metadata specifications to exchange data.

Personalization is a powerful mechanism to assist users. The architecture enables personalization through recommendations and apps, e.g. (i) recommendations will be provided based on tracked user activities, (ii) learning analytics apps support awareness and self-reflection and (iii) scaffolding apps assist students. Through the separate learning analytics and recommendation component, user traces can be tracked over all portal components, data privacy can be ensured by anonymization and various implementations are possible applying different data processing techniques. The separate user management component facilitates the integration with various authentication systems of large educational institutions.

To really create an impact, it will be very important to attract many teachers and lab owners to use the portal. That is why we see the Go-Lab portal as a two-sided marketplace where ILS are provided by suppliers (lab-owners and teachers), and used by customers (teachers). Such markets typically benefit from network effects [13], that is, the more suppliers the market attracts, the more customers will join. In turn, more customers will attract more suppliers, creating a positive feedback loop. The challenge with such systems is to jump start this process. On the one hand, we believe that empowering teachers by allowing them to easily customize and republish ILS will lead to a greater supply. On the other hand, we are including STEM teachers from the start of the project on a large scale (100 teachers the first year up to 1000 teachers after three years) through our partnership with organizations such as the European Schoolnet[16] leading to a large user base.

Acknowledgment. This research was partially funded by the European Union in the context of the ROLE (grant no. 231396) and Go-Lab (grant no. 317601) projects under the ICT theme of the 7th Framework Programme for R&D (FP7).

[16] European Schoolnet, `http://www.eun.org`

References

1. Avison, D., Shah, H.: The Information Systems Development Life Cycle: A First Course in Information Systems. McGraw-Hill Publishing Co. (1997)
2. Bogdanov, E., et al.: A social media platform in higher education. In: Proc. of the IEEE Global Engineering Education Conf. (EDUCON), 2012. 1–8 (2012)
3. Bohl, O., Scheuhase, J., Sengler, R., Winand, U.: The sharable content object reference model (SCORM) – a critical review. In: Proc. of the Int. Conf. on Computers in Education. 950–951 vol.2 (2002)
4. Maier, C., Niedersttter, M.: Lab2go A repository to locate online laboratories. Int. Journal of Online Engineering (iJOE) Vol 6(No. 1) (2010)
5. Dahrendorf, D., Dikke, D., Faltin, N.: Sharing personal learning environments for widget based systems using a widget marketplace. In: Proc. of the PLE Conf. (2012)
6. Lowe, D. et al.: LabShare: Towards a national approach to laboratory sharing. Proc. of the 20th Australasian Assoc. for Engineering Education Conf. (2009)
7. Drucker, P.: The age of: Guidelines to our changing society. Transaction books, Transaction Pubs. (1992)
8. Garcia-Zubia, J., Lopez-de Ipina, D., Orduna, P., Hernandez-Jayo, U.: Experience with weblab-deusto. In: the IEEE Int. Symp. on Industrial Electronics. vol. 4, 3190–3195 (2006)
9. Gillet, D., de Jong, T., Sotirou, S., Salzmann, C.: Personalised Learning Spaces and Federated Online Labs for STEM Education at School: Supporting Teacher Communities and Inquiry Learning. In: Proc. of the 4th IEEE Global Engineering Education Conference (EDUCON). 769–773. IEEE (2013)
10. Harward, V. et al.: The iLab shared architecture: A web services infrastructure to build communities of internet accessible laboratories. Proceedings of the IEEE 96(6), 931–950 (2008)
11. de Jong, T., Linn, M.C., Zacharia, Z.C.: Physical and virtual laboratories in science and engineering education. Science 340(6130), 305–308 (2013)
12. Minner, D.D., Levy, A.J., Century, J.: Inquiry-based science instruction - what is it and does it matter? Results from a research synthesis years 1984 to 2002. Journal of Research in Science Teaching (2009).
13. Parker, G.G.: Two-sided network effects: A theory of information product design. Management Science Vol. 51(No. 10), 1494–1504 (2005)
14. NISO: Understanding Metadata. National Inf. Standards Org. Press (2004)
15. Richter, T., Boehringer, D., Jeschke, S.: LiLa: A European project on networked experiments. In: Automation, Communication and Cybernetics in Science and Engineering 2009/2010, 307–317. Springer Berlin Heidelberg (2011)
16. Rocard, M. et al.: Rocard report: Science education now: a new pedagogy for the future of Europe. Tech. rep., European Commission (2006)
17. Singer, S., Hilton, L., Schweingruber, H.: America's Lab Report: Investigations in High School Science. The National Academies Press (2005)
18. Richter, T., Grube, P., Zutin, D.: A standardized metadata set for annotation of virtual and remote laboratories, In: The IEEE Int. WS on Multimedia Tech. for E-Learning (MTEL) (2012)
19. Turnitsa, C.: Extending the levels of conceptual interoperability model. Proceedings IEEE 2005 Summer Computer Simulation Conference, IEEE CS Press (2005)
20. Vogel, O., Arnold, I., Chughtai, A., Kehrer, T.: Software Architecture – A Comprehensive Framework and Guide for Practitioners. Springer (2011)
21. Vozniuk, A., Govaerts, S., Gillet, D.: Towards portable learning analytics dashboards. In: Proc. of the 13th IEEE Int. Conference on Advanced Learning Technologies (2013)
22. Zutin, D., Auer, M., Maier, C., Niederstatter, M.: Lab2go – a repository to locate educational online laboratories. In: Education Engineering (EDUCON), 2010 IEEE. 1741–1746 (2010)