

Go-Lab Specifications of the Learning Analytics, Scaffolding Services, and Add-on Services -Final

Sven Manske, Yiwei Cao

▶ **To cite this version:**

Sven Manske, Yiwei Cao. Go-Lab Specifications of the Learning Analytics, Scaffolding Services, and Add-on Services -Final. [Research Report] Go-Lab Project. 2015. hal-01275002

HAL Id: hal-01275002
https://telearn.hal.science/hal-01275002

Submitted on 16 Feb 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Go-Lab

Global Online Science Labs for Inquiry Learning at School

Collaborative Project in European Union's Seventh Framework Programme
Grant Agreement no. 317601

Deliverable D4.6

Specifications of the Learning Analytics, Scaffolding Services, and Add-on Services - Final

Editors	Sven Manske (UDE)
	Yiwei Cao (IMC)
Date	31st July, 2015
Dissemination Level	Public
Status	Final

©2015, Go-Lab consortium

The Go-Lab Consortium

Beneficiary Number	Beneficiary Name	Beneficiary short name	Country
1	University Twente	UT	The Netherlands
2	Ellinogermaniki Agogi Scholi Panagea Savva AE	EA	Greece
3	École Polytechnique Fédérale de Lausanne	EPFL	Switzerland
4	EUN Partnership AISBL	EUN	Belgium
5	IMC AG	IMC	Germany
6	Reseau Menon E.E.I.G.	MENON	Belgium
7	Universidad Nacional de Educación a Distancia	UNED	Spain
8	University of Leicester	ULEIC	United Kingdom
9	University of Cyprus	UCY	Cyprus
10	Universität Duisburg-Essen	UDE	Germany
11	Centre for Research and Technology Hellas	CERTH	Greece
12	Universidad de la Iglesia de Deusto	UDEUSTO	Spain
13	Fachhochschule Kärnten - Gemeinnützige Privatstiftung	CUAS	Austria
14	Tartu Ulikool	UTE	Estonia
15	European Organization for Nuclear Research	CERN	Switzerland
16	European Space Agency	ESA	France
17	University of South Wales	USW	United Kingdom
18	Institute of Accelerating Systems and Applications	IASA	Greece
19	Núcleo Interactivo de Astronomia	NUCLIO	Portugal

Contributors

Name	Institution
Sven Manske, Tilman Göhnert, Tobias Hecking, H. Ulrich Hoppe	UDE
Anjo Anjewierden, Lars Bollen, Jakob Sikken	UT
Yiwei Cao, Nils Faltin	IMC
María Jesús Rodríguez-Triana, Sten Govaerts, Andrii Vozniuk, Wissam Halimi, Christophe Salzmann	EPFL
Danilo Garbi Zutin	CUAS
Pablo Orduna	UDEUSTO
Denis Gillet (Reviewer)	EPFL
Ton de Jong (Reviewer)	UTwente

Legal Notices

The information in this document is subject to change without notice. The Members of the Go-Lab Consortium make no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Members of the Go-Lab Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material. The information and views set out in this deliverable are those of the author(s) and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.

Executive Summary

This deliverable describes the final specification of learning analytics, scaffolding, and add-on services of Go-Lab. It follows the initial specification reported in D4.2 (M18) and reflects the development and evaluation of the release in D4.4 (M24). It serves as a guideline for final release of the learning analytics, scaffolding, and add-on services in D4.8 (M36).

This deliverable consists of two major parts: (i) the learning analytics and scaffolding services and (ii) the add-on services including a booking system and a tutoring platform (which was called the Bartering Platform in D4.2).

The learning analytics and scaffolding services consist of a rich back and several options to develop new learning analytics apps according to the specified infrastructure and architecture. To support the two main stakeholders of learning analytics in Go-Lab, namely teachers and learners, a teacher dashboard and several apps, e.g. to support learners' reflection, will be presented in this deliverable. Participatory design activities, particularly framed by the Go-Lab Summer School 2015, have been used to evaluate initial versions of the learning analytics apps to gather useful feedback aligned to the stakeholders.

The booking system offers Go-Lab remote labs an appropriate booking service using a calendar managed by lab owners. The tutoring platform helps build up a virtual community where teachers could share their expertise in inquiry learning with online labs and help each other grow their teaching skills.

Table of Contents

1 Introduction **8**

2 Learning analytics and scaffolding services **9**
2.1 Introduction to learning analytics and scaffolding services 9
2.2 User scenarios 10
2.3 Requirements 11
2.3.1 Composition of micro-services for LA 13
2.3.2 Requirements for the Go-Lab learning analytics infrastructure 13
2.3.3 Requirements for graceful scaling of the LA features . . . 13
2.4 Architecture and specifications 13
2.4.1 Components overview 14
2.4.1.1 Analytics services 14
2.4.2 Data access and retrieval in the Go-Lab portal 15
2.4.2.1 Activity tracking in the Go-Lab Repository 16
2.4.2.2 Activity tracking in the inquiry learning spaces platform 16
2.4.2.3 Data Access for analytics services and components 17
2.4.2.4 Development server infrastructure 17
2.5 Development of new learning analytics apps for Go-Lab 18
2.5.1 Starter App: Skeleton of a learning analytics app 19
2.5.2 Example 1: Development of an app using on-demand services and the starter app 19
2.5.3 Example 2: Development of a realtime app using the feedback loop 19
2.5.4 Example 3: Development of micro services with the analytics workbench 20
2.5.5 Focus on advanced LA apps for teachers and students . . 20
2.6 Advanced LA apps for students 20
2.6.1 Example 1: Reflection tools 20
2.6.2 Example 2: My learning process 21
2.6.3 Example 3: Concept Map Aggregation 24
2.7 Advanced LA apps for teachers and the teacher dashboard . . . 24
2.7.1 Dashboard templates 26
2.7.2 Indicators and apps 26
2.7.3 Example 26
2.8 User evaluation of initial versions of apps for teachers and students 27
2.8.1 Evaluation of LA apps for students 28
2.8.2 Evaluation of LA apps for teachers 29
2.9 Specification of the Golabz recommender system 33
2.9.1 Data sources 33
2.9.2 Design of the recommender system 34
2.9.2.1 Recommendations for apps 34

2.9.2.2 Recommendations for online labs 35
2.9.3 Technology specification 36

3 Add-on services **38**
3.1 Introduction to Go-Lab add-on services 38
3.2 Go-Lab Booking System 39
3.2.1 Requirements for the Go-Lab booking system 40
3.2.2 Integration specification of the Go-Lab Booking System . 42
3.3 Go-Lab Tutoring Platform 44
3.3.1 Use scenario of the Go-Lab Tutoring Platform 45
3.3.2 Requirements of the Go-Lab Tutoring Platform 45
3.3.3 Components of the Go-Lab Tutoring Platform 47
3.3.3.1 Google services integration 47
3.3.4 Support teachers virtual community building 48

4 Conclusion **51**
References 53

1 Introduction

This deliverable reports on the revised specification of learning analytics, scaffolding, and add-on services. To support readability, we structure the deliverable in two parts as in previous deliverables: (i) learning analytics & scaffolding and (ii) add-on services.

According to the initial specifications, we have developed the initial releases and collected some first-hand development experiences collaborated with colleagues from Work Package 3. Based on the available evaluation of the first releases, we refine the requirements, architecture, and use scenarios of the final specifications in this deliverable D4.6.

In addition, the feedback from EC project reviewers of Year 2 has been taken into account. This includes the comments and recommendations regarding the early deployment of the tutoring platform as well as comments related to Learning Analytics. Additionally, we have refined the implementation plan in such a way as to include the usage scenarios together with other Go-Lab work packages, including WP2, WP3 and WP6.

This deliverable D4.6 extends and partially revises the initial specifications in D4.2. Wherever appropriate important prior decisions from D4.2 are briefly summarised and relevant changes are noted in this deliverable. The future deliverable D4.8 will describe concrete agents, services and apps and their respective implementations in line with the specifications defined in this deliverable.

2 Learning analytics and scaffolding services

2.1 Introduction to learning analytics and scaffolding services

Siemens (2012) defines learning analytics as “the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimising learning and the environments in which it occurs”. To achieve these goals, learning analytics brings together different fields, i.e. business intelligence, web analytics, educational data mining and recommender systems (Ferguson, 2012). Methods comprise (1) content analysis of produced artefacts by learners (such as concept maps (de Jong et al., 2010; Hoppe et al., 2012; Clariana et al., 2013) and texts (De Wever et al., 2006; Liddo et al., 2011; Southavilay et al., 2013)), (2) learner behaviour analysis (Zaıane & Luo, 2001; Facca & Lanzi, 2005; Duval, 2011) and (3) social network analysis (Laat et al., 2007; Zhang et al., 2007; Harrer et al., 2007). A major challenge, also for the Go-Lab project, is to combine these approaches in a flexible infrastructure in order to achieve a productive synergy (Suthers et al., 2013).

The Go-Lab portal as an environment for various kinds of inquiry-based learning activities generates many types of data, including traces of interaction with the system and results of the learning process. The learning analytics services make use of such data to provide analytical information for the Go-Lab portal (D5.2) and add-on services (section 3) in order to foster awareness, create individual scaffolds for students as well as information for teachers supporting the monitoring of learning activities and better informed decision making.

The focus of the analytics and scaffolding services in Go-Lab is on interaction and content analysis, since social relations are not explicit in the environment. Interaction analysis is based on the action logs collected in the LA backend through a specific tracking agent (AngeLA). In turn, content analysis refers to learner-generated artefacts (e.g. concept maps or hypotheses) that are accessible through the ILS’s data repository (the "vault").

For a better overview and distinction of the Go-Lab learning analytics services it is appropriate to distinguish the following three levels (outside-in): The outmost perspective addresses the portal as a whole. On this level, the learning analytics services can help teachers to find appropriate resources, for example through the recommendation of apps, labs and ILS templates. Technically this can be supported by web analytics techniques. The second level is related to single inquiry learning spaces (ILS). Support on this level includes monitoring support for teachers, providing a visual account of the students’ activities in an ongoing session. Similar information can be given to students to support reflection. This level also includes the analysis of action log protocols comprised of time stamped events like the access of a resource or app usage. The innermost level refers to the analysis of the student actions inside particular apps and the thereby produced data (actions or products). This is an important source of student scaffolding, yet it depends on the internal specification of the app.

Scaffolding mechanisms assist learners in tasks that they cannot solve without

guidance. Scaffolding is based on previous analysis and modelling of learner behaviour. Typical scaffolding mechanisms are immediate feedback (e.g. with example recommendations) or the provision of more context information in an app. Scaffolding services for learners rely on information about the labs, their users and corresponding user activities as well as the subject domain of the lab. Therefore scaffolding apps are dependent on the lab metadata scheme developed in WP5 as well as the smart device and gateway for remote labs (see D4.1).

For all types of learning analytics it is beneficial that the results be accessible in the current learning context and not only in separate spaces (Harrer & Göhnert, 2015). This facilitates the usage of the learning analytics results as they are offered in the context of the learning and teaching process and thus can also be connected more easily. This means there is not only the challenge to define and implement appropriate learning analytics tools but also the challenge to integrate these with the learning and authoring platforms. One possible approach is to embed the results of external analysis tools (of possibly more general type) into the learning platforms to make the results of the analysis process directly accessible in the context of the platform. Technical solutions and architectural models for this approach have been developed by (Göhnert et al., 2014) and (Manske et al., 2014), the latter explicitly contextualised in the framework of Go-Lab.

2.2 User scenarios

The following user scenarios highlight how the stakeholders can benefit from the learning analytics services within the Go-Lab portal and apps. In D4.2, the following scenarios have been described:

- Scenario LA-1: Learning analytics for the Go-Lab Repository.
- Scenario LA-2: Learning analytics on the ILS platform level.
- Scenario LA-3: Learning analytics in apps.

Scenario LA-1 covered recommendations of resources in the Go-Lab Repository, LA-2 is focused on LA apps for supporting teachers in supervision, which is realized through a teacher dashboard. Complementary, scenario LA-3 described individual learner support through learning analytics apps which are embedded in the ILS platform. These three scenarios (the implementation will be detailed in D4.8) described the usage of learning analytics to support teachers and students on different parts of the Go-Lab ecosystem, namely in the Go-Lab portal, the Go-Lab Repository, the ILS platform and through single OpenSocial apps. Besides these stakeholders, we extended the set of scenarios to involve the other stakeholders of learning analytics for Go-Lab. Scenarios LA-4 demonstrates how an analyst uses the analytics workbench in Go-Lab to create a micro-service to be deployed and published without any programming. The scenario LA-5 describes the use case of the development of learning analytics apps and services for project members and researchers in the field of LA.

Scenario LA-4: Composition of micro-services to be embedded as an app. Michael is an analyst who knows a lot about different analysis techniques for analysing learners' activities but is not very experienced in programming. However he has developed an idea for an analysis of learner activities in the Go-Lab context that might help teachers in guiding the learning process. In order to create a learning analytics app out of his idea he uses the authoring mode of the Analytics Workbench. The Analytics Workbench offers the possibility to build complex analysis workflows from basic building blocks, which encapsulate individual steps of an analysis process like data import, data transformation, analysis techniques, and data visualisation. After having built a first version of his analysis workflow, Michael runs a test based on input data taken from an ILS he knows about. Based on the results, he refines the workflow and tests it again until he is satisfied with the implementation of his initial analysis idea. In order to make the analysis available for users of the Graasp portal, he uses the publishing service of the workbench. This allows him to build an analysis service based on the configured workflow and export it in the form of an OpenSocial gadget that is wrapping the calls of the service and allows visualising the results. The service is created in such a way that if the gadget is connected to an ILS it automatically uses the logs of that ILS as input data. That way Michael has created a learning analytics app in the form of OpenSocial gadget that can be connected to any ILS.

Scenario LA-5: Development of LA services and apps. Researcher Albert had an idea about a new visualisation to support learners in reflecting about their progression in inquiry learning scenarios. He wants to incorporate artefacts and actions to display a general measurement of the level of activity as well as to feed back the progression on artefacts (such as concept maps and hypotheses) to the students. For this purpose, he decides to develop a specific learning analytics app for Go-Lab.

After Albert plans his app, he starts with the development. He identifies two main components he needs to develop: first, he needs an OpenSocial app for the client to be embedded in the ILS and exposed to the Go-Lab portal. This app will be aware of the context it runs in. Second, this app needs to communicate with a service to be embedded inside the LA infrastructure. The service component has access to the artefacts with contextual information and the action logs of the students through the data access API of the infrastructure. Alber bases his app on the Go-Lab "Starter App", a skeleton for LA apps that provides access to the Go-Lab APIs and contextualized information. He embeds his innovative visualisation into the Starter App and places web service calls to retrieve the data model from his service that will be deployed within the LA infrastructure.

2.3 Requirements

In D4.2, the main stakeholders and their information needs have been described as a basis for formulating functional requirements. The emphasis in the initial

specification has been put on students, teachers and lab-owners as stakeholders and, additionally, on the types of data that these actors produce on different levels of the Go-Lab ecosystem. Now, D4.6 puts the focus on the actual development and implementation of learning analytics apps and services for Go-Lab. This involves on the one hand the creation of micro-services by expert analysts (scenario LA-4) and the (software-) development of agents, services and apps for learning analytics (scenarios LA-5). A global view on learning analytics involving an integrated perspective on the different data sources will be beneficial for researchers in the Go-Lab consortium in general and particularly for Community Support (WP7).

To meet the requirements expressed and exemplified in the scenarios listed above several LA services have been implemented in Go-Lab:

- The Go-Lab repository (Golabz.eu) which is implemented in Drupal has been extended with ElasticSearch[1] for offering labs and apps recommendations based on the description of these items. The Repository is also enhanced with Piwik, Google Analytics, and Drupal web statistics module to get metrics on visits by teachers and lab owners in order to assess the dissemination of the project and to highlight the most frequently accessed or exploited resources. These features will be described in D5.6.
- The ILS Platform (Graasp.eu) which is implemented with MEAN.js supports the Activity Stream standard to track and share with the LA backend (when AngeLA is enabled) students activities traces gathered in ILS. Activity Stream has been chosen at the beginning of the project as an open data exchange format to enable interoperability between the Go-Lab platforms. Contextual users are also introduced in Graasp for a better tracking scheme which properly enforce privacy. Finally, Google Analytics has been activated in this platform also to help getting dissemination metrics for the project. These features will be described in D5.6.
- The OpenSocial apps which are offered by Go-Lab to provide scaffolding or interfaces to online labs can all consume directly the activity stream corresponding to the ILS in which they are integrated, thanks to the ILS library. Hence, they can elaborate and provide their own analytics to their users. Such apps are listed in D5.5 (like *File Drop* which is showing the content produced by students and stored in the Vault or *Student time spent* which displays a table with the time spent for each student in each phase of an Inquiry Learning Space).
- Advanced learning analytics apps (referred hereafter as apps integrating a feedback loop or using analytics services) which are offered by Go-Lab to provide high-level feedback to teachers and learners relying on the dedicated LA backend. They benefit from the rich analytics infrastructure, e.g., for analyses across different contexts, and use higher level APIs for aggregations and ex-post analysis. As they are the only ones to be able to

[1] ElasticSearch, `http://www.elastic.co`

consume traces gathered in various ILS using similar apps (when AngeLA is enabled), they integrate a special development scheme and a policy to enforce privacy. **This deliverable mainly describes the infrastructure for such advanced apps**.

2.3.1 Composition of micro-services for LA

Scenario LA-4 describes a situation, where an analysis expert but not experienced programmer wants to create a learning analytics app, test and refine it, and make it publicly available for use by others. In order to do this, he needs a tool or tools that support him in all steps of that process. As a first step, the tools must allow defining an analysis process without programming work. In the next step, that workflow needs to be executed for testing purposes. Ideally, the tools also support the analyst in the transition between workflow definition, testing it, and editing it again for refinements. Finally, the tools need to offer a possibility for creating an analysis app that can be published and then used by the creator and/or others. Ideally this app would allow embedding the analysis results into the learning platform.

2.3.2 Requirements for the Go-Lab learning analytics infrastructure

Scenario LA-5 described the use case of a Go-Lab researcher and developer, who wants to program her own learning analytics app to be deployed to the learning analytics server. Therefore, the server needs to provide some flexibility, to develop under different paradigms and to serve different modalities of the apps (realtime or on-demand services, feedback loop) as well as high-level abstractions and APIs. On the other hand, access to data sources should not be limited in the same way as for general OpenSocial apps, not to prevent innovative analytics approaches, methods and visualisations on beforehand. Such openness, particularly with the connected data sources, which potentially contains sensitive learner data, which are usually not exploited to the outside. To circumvent this, a privacy policy is needed, to ensure that every Go-Lab developer of the consortium does not violate privacy with the development of new learning analytics apps. This affects the deployment of such apps, which is generally controlled through the Go-Lab consortium and shall not allow third-party app developers to expose sensitive data.

2.3.3 Requirements for graceful scaling of the LA features

A main prerequisite for learning analytics, particularly based on activity traces, is the existence of action logs. To let the teacher be in control of her students' privacy, she is allowed to turn off the tracking through AngeLA. As a consequence, learning analytics apps won't function regularly as they are lacking appropriate data of the learners' traces. Therefore, several ways to deal with this need to be specified.

2.4 Architecture and specifications

This section describes a specification of the Go-Lab learning analytics infrastructure. Based on the requirements above, the architecture is intended to be

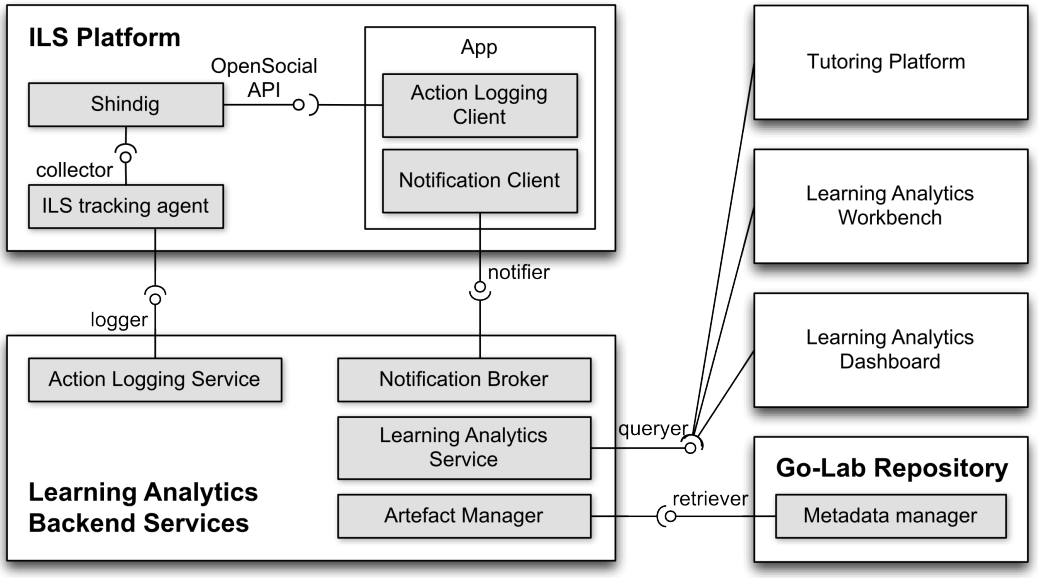


Figure 1. Components of the learning analytics infrastructure.

modular, and open for extensions. First an overview on the general services and components is given. After that specific solutions for an agent based architecture, data retrieval and the generation of user feedback are given. This is followed by an initial specification of action logging and notification formats. Applications that use the described infrastructure and privacy mechanisms are presented at the end of this section. This section highlights the functionality implemented for the learning analytics and scaffolding services based on the specifications in D4.2. As a follow up of the initial specifications in D4.2, we present the different models of analytics services specified for learning analytics in Go-Lab.

2.4.1 Components overview

Figure 1 depicts an overview of the learning analytics infrastructure. It can be divided in server-side services, namely the back-end services, and client side services in the ILS platform in the form of a user tracking agent and client APIs for action logging and notifications.

2.4.1.1 Analytics services

The analytics services are organised in three different categories, serving different developing paradigms and programming models. The development with respect to these models is described in section 2.5 on behalf of examples. The categories induce a scope and execution mode of the services:

- Realtime,
- On-demand, and
- Micro services.

Realtime services are used for immediate interventions, such as concept recommendations or adaptations of scaffolds. The development paradigm is based upon a shared memory, which acts as a short-term memory for realtime analysis for connected agents. These interventions are realized through a feedback loop (cf. D4.2): action logging of learners' activities results in traces that are usually analysed by agents (cf. D4.4). The shared memory stores all short-term analysis information such as action logs, temporary and final analysis results, and notifications which are then collected by a notification agent. This is useful to realize interventions: notification messages are created and sent out through the notification broker to the connected clients providing a back channel in the feedback loop.

On-demand services are usually stateless web services which are called actively when needed ("on-demand") by client apps. This induces an execution mode, where client apps present static models that are not aggregating updates frequently. Typical examples involve aggregating services, especially when the aggregations are complex, e.g. across different ILS. The services of this category offer well-defined REST interfaces.

Micro services are characterized through their scope which is points to a usage on a personal level, e.g., an analyst creating an app for his own purposes without exposing it through official channels such as the Go-Lab portal. This part of the infrastructure specification is different to the rest as it is based on the analytics workbench, which is an existing and independent system, but adapted and integrated to the Go-Lab LA architecture (cf. D4.4, section 2.3). The analyst specifies the workflow with a graphical language based on the pipes and filters metaphor, where she connects analysis components and filters to specify workflows. If this results in a visualisation, an app can be generated from this editor, which will be deployed and hosted in the LA infrastructure, being accessible through an URL. This URL can be used to embed the app in the Go-Lab learning environment.

2.4.2 Data access and retrieval in the Go-Lab portal

In the requirements analysis of D4.2, different information needs and sources of data have been identified. The different data sources in Go-Lab need to be exploited for learning analytics. As a characteristic, the different data sources are heterogeneous in their nature, and need to be integrated to create a a holistic view on Go-Lab to draw meaningful conclusions about the usage and its users. As an example, this could be a trajectory of inquiry learning spaces, starting from a teacher in Golabz, creating a space from a lab, while students use the space. Activity metrics can be used to identify these prototypical examples. As a result, such teachers who adopt the Go-Lab approaches well, can be selected as "core-teachers" to foster the community building around WP6 and also be utilized to strengthen a sustainable teacher-community beyond this project.

2.4.2.1 Activity tracking in the Go-Lab Repository

As mentioned, Go-Lab Repository users will often not be logged in, so detailed user information will not be available while tracking. Therefore, we will rely on existing web analytics solutions. We selected Piwik[2] because it is open source, free and allows us to store the tracked data on our own servers to ensure better privacy than with cloud-based solutions like Google Analytics.[3]. The REST endpoint of Piwik will be exploited by the LA backend to enrich the traces of teachers using Graasp. Browsing the pages of different online labs or apps before the creation of an inquiry learning space can be seen as a valuable, implicit information.

2.4.2.2 Activity tracking in the inquiry learning spaces platform

The semantics, formats and APIs for activity tracking have been specified in D4.2 and the implementation described in D4.4. Entities to be logged are process- or product-oriented. Process-oriented entities are action logs representing the trace of a user in the portal, while product-oriented enrich this trace by artefacts, e.g. user-generated content such as learning objects. These artefacts capture - in contrast to action logs - the state of an artefact.
As a component to foster privacy awareness, an explicit tracking agent is available in Graasp. The Learning Analytics Tracking Agent (AngeLA) functionality consists of the following aspects:

1. **Tracking permission management:** AngeLA aggregates activity logs only from the spaces where it is a member. This provides an easy to use way to manage user tracking permissions. To enable the activity tracking in a space a user just needs to invite AngeLA to this space. Figure 2 shows a screenshot of such a space with AngeLA as a member. When AngeLA is removed from the space, the tracking is disabled for that space. This behaviour is intuitive for the teacher, since the teacher is expecting all members of a space to be aware of activities happening inside.
2. **Cross-space activity data collection:** AngeLA continuously aggregates activity logs of the Graasp users across the spaces where it is a member. The activities are aggregated into a single activity stream. Graasp access rights are enforced to enable access to this stream or not, which corresponds to confining information in the ILS it belongs, i.e. showing it only to its authorized members.
3. **Data transmission to the Learning Analytics Backend:** All the activity records collected are sent to the LA backend for further processing. The Activity Streams format is used to represent the actions during the transmission.

The functionality presented above was implemented and deployed on the production version of Graasp. To achieve this, the following has been accomplished (1) implementing the default policy for placing AngeLA into the space and (2)

[2]Piwik, `http://piwik.org/`
[3]Google Analytics, `http://www.google.com/analytics/`

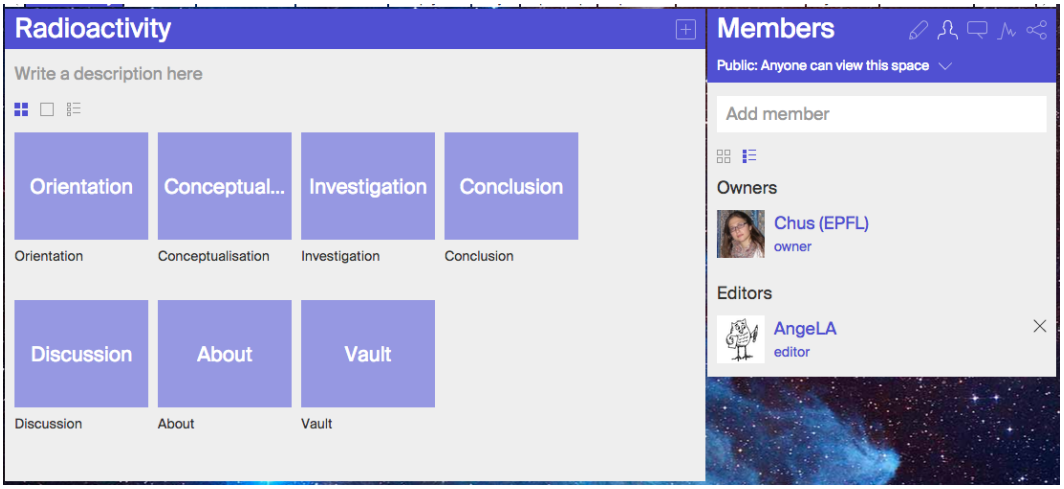


Figure 2. A screenshot of the Learning Analytics Tracking Agent in a space.

collecting and sending the student and teacher actions happening in the Inquiry Learning Space. The latter requires the identities of anonymous students to be represented in Graasp, which has been realized through the implementation of contextual users.

2.4.2.3 Data Access for analytics services and components

This layer is on top of the data warehouse in the described architecture. It provides a retrieval layer for the entities of data in the learning analytics infrastructure, namely action logs and artefacts. The purpose is to have an abstraction over the database layer and to integrate the different data sources into a standardized access layer. The development of learning analytics apps, agents and services benefits from this access layer as it provides a simple and unified API for the retrieval and querying of data from the data warehouse. Figure 3 depicts the data flow between the components of the learning analytics infrastructure, particularly the data access provided for the different services by the data warehouse API.

2.4.2.4 Development server infrastructure

The Go-Lab learning analytics infrastructure follows a two staging deployment process, consisting of a development and a production server. The data is mirrored from the production server to the development server to guarantee a similar functionality through action logging, although the ILS platform is only connected with a single endpoint. This mirroring is handled in the same way as the forwarding of action logs to the other subsystems such as the data warehouse or the shared memory (cf. D4.4). In a similar way, further subsystems systems can be connected to this infrastructure, for example a teacher dashboard for monitoring.

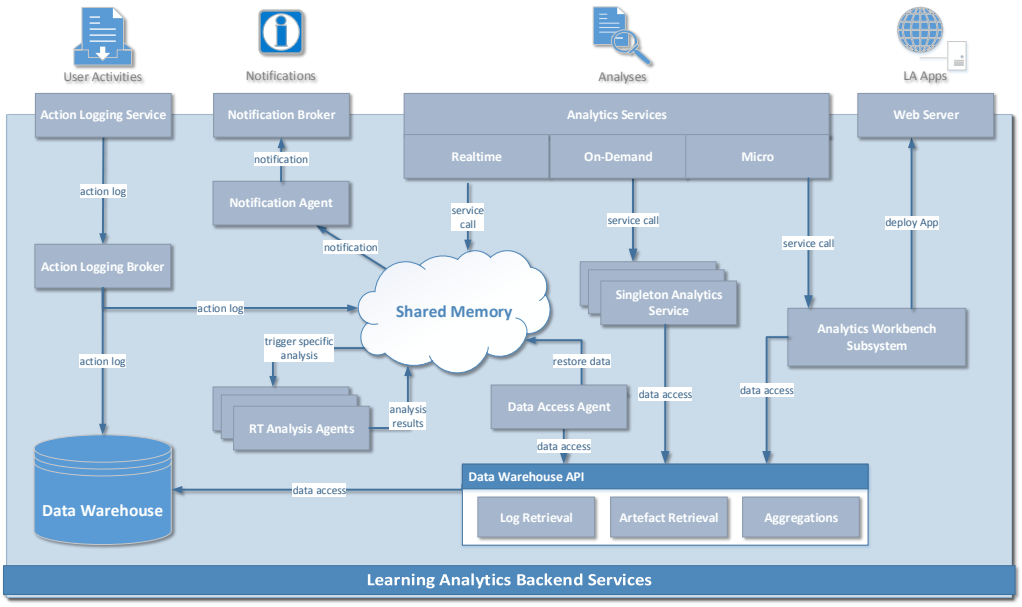


Figure 3. Information flow between components in the learning analytics infrastructure.

2.5 Development of new learning analytics apps for Go-Lab

In this section, we describe how the Go-Lab learning analytics infrastructure can be used to develop new learning analytics apps and services. Basically, there are two ways to organize and realize the development of a new app, which have been described in the scenarios LA-4 and LA-5:

1. Through software development of agents and services (depicted in examples 1 and 2 in sections 2.5.2 and 2.5.3, both covering scenario LA-5),
2. Graphical composition through the analytics workbench (example 3 satisfying LA-4 in section 2.5.4).

The analytics workbench provides facilities to create and deploy micro-services for Go-Lab. The advantage is in the ease of use - which does not need any programming or extra deployment mechanisms. The workbench has its own, internal deployment for the apps and connects to the Go-Lab data sources through an integration layer with access to the data warehouse.
The second option involving programming has its advantages with a tighter coupling and more degrees of freedom. To integrate the new services into the existing infrastructure, more efforts are needed, particularly for their deployment. This also allows for establishing a control mechanism on part of the platform, which restricts malicious services to be deployed, that could potentially expose sensitive learner data. The Go-Lab consortium controls this deployment based on its privacy policy.
The documentation for the development of learning analytics apps and services can be found on the home page of the learning analytics development server on `http://golab-dev.collide.info/`. It contains a collection of development

resources that have been accumulated across the past deliverables and the existing API documentations. The documentation explains both the development of client apps for learning analytics based on the Go-Lab integration libraries, and the backend development of analytics services and agents using the feedback loop.

2.5.1 Starter App: Skeleton of a learning analytics app

To easily develop new learning analytics components for Go-Lab, a skeleton of an app with simple functionality has been provided. This app makes use of the Go-Lab learning analytics services in this infrastructure and uses the Go-Lab libraries for action logging, storage of artefacts, etc. This stands on the one hand as a prototypical example, and on the other hand it demonstrates best practices for the app development.

2.5.2 Example 1: Development of an app using on-demand services and the starter app

Following this approach, the app consists of mainly two parts: (1) a client-side app to be embedded into the ILS platform, and (2) a service to be deployed with the LA infrastructure. We demonstrate this development with an app that aggregates action logs for the ILS it runs in. The interaction paradigm will be: data is loaded on demand through a web service. The response contains the necessary data to be rendered in the client app.
One of the major difficulties for the client-app development is the effort of integration, which is necessary for this approach. Go-Lab, particularly the ILS platform, supports the embedding and rendering of OpenSocial gadgets (cf. D5.1). The set of Go-Lab libraries, for handling artefact storage, contextualized metadata, action logging, notifications, internationalisation, etc., provide means of integration on top of the OpenSocial API (D4.2 and D4.4). The starter app is a skeleton and example for a simple app that uses all these libraries. Therefore, it provides quick access to the integration libraries demonstrating their usage in a user interface. This app can be already embedded into Graasp, but adaptations to the use case, for example the actions to be logged, need to be performed as one of the main integration tasks.
To create the server component, a simple script can be written. The current implementation of the main infrastructure is based on node.js and express as a REST middleware. New web services can be integrated into a service router, which redirects the calls and executes the associated script. In the case of action statistics, an aggregation of action logs is needed. A data warehouse API, which abstracts and encapsulates the logic to retrieve and aggregate data from the data warehouse, can be used. The service needs to be integrated tightly into the service infrastructure and must be deployed together with all the other static analytics REST web services.

2.5.3 Example 2: Development of a realtime app using the feedback loop

The development of a learning analytics app with intermediate feedback following this approach consists of similar parts, namely a client-side part using the Go-Lab libraries, at least notifications and action logging. These two parts are

necessary to realize the feedback loop (cf. D4.2). In D4.4, example agents have been described: the concept mapping agent is a good example for this paradigm: he manages and updates continuously a model of the learner, which is kept in the agent's short-term memory. The concept recommendations are then propagated through the notification channel to the client-side. On the client-side, the concept mapper itself registers to handle this specific kind of notifications to be rendered. In a similar approach, any client-side app could process these notifications based on the initial specifications described in D4.2.

2.5.4 Example 3: Development of micro services with the analytics workbench

The use case of this example has been described in scenario LA-4 and the basic, underlying architecture, particularly the one of the analytics workbench has been described in D4.4. The usage example in section 2.3.5 of D4.4 shows the creation of the concept map aggregation app.

2.5.5 Focus on advanced LA apps for teachers and students

In this deliverable, we mainly focus on "advanced learning analytics apps", which are based on the architecture described above and are using higher level abstraction, e.g. for action log aggregations or using the feedback loop. The description of all advanced learning analytics apps will be subject of the follow-up deliverable D4.8. In the sequel, we provide an overview of a few examples highlighting facets of the infrastructure.

2.6 Advanced LA apps for students

Whereas teachers will be provided with specific subspaces offering a set of learning analytics in the form of dashboard (see next section), apps for students will be embedded with other inquiry learning phases (typically with the reflection phase). Accordingly the first example is an app supporting reflection. It uses the agent-based approach realizing the (technical) feedback loop, whereas the second example ("my learning process") uses advanced services to retrieve representations of learning process models. In contrast, the third example ("concept map aggregation") is based on the micro-service approach.

2.6.1 Example 1: Reflection tools

Two reflection tools have been implemented based on the LA infrastructure. For both tools, the server-side component is based on the Tuple Space approach, with agents processing action logs. The time spent reflection tool presents the student with an overview of the time spent percentages in the inquiry phases of an ILS (see Figure 4). This tool is intended to be included in an ILS by the teacher (most often in the Discussion phase). The teacher can configure the tool by setting norm percentages (shown as black bars in the figure). Students can then reflect on their personal time spent (blue bars) by answering reflection questions configured by teachers. The second reflection tool shows a time-line with the ILS phases and when the student visited the phases (see Figure 5). This tool helps students reflect on the order in which they used the phases and

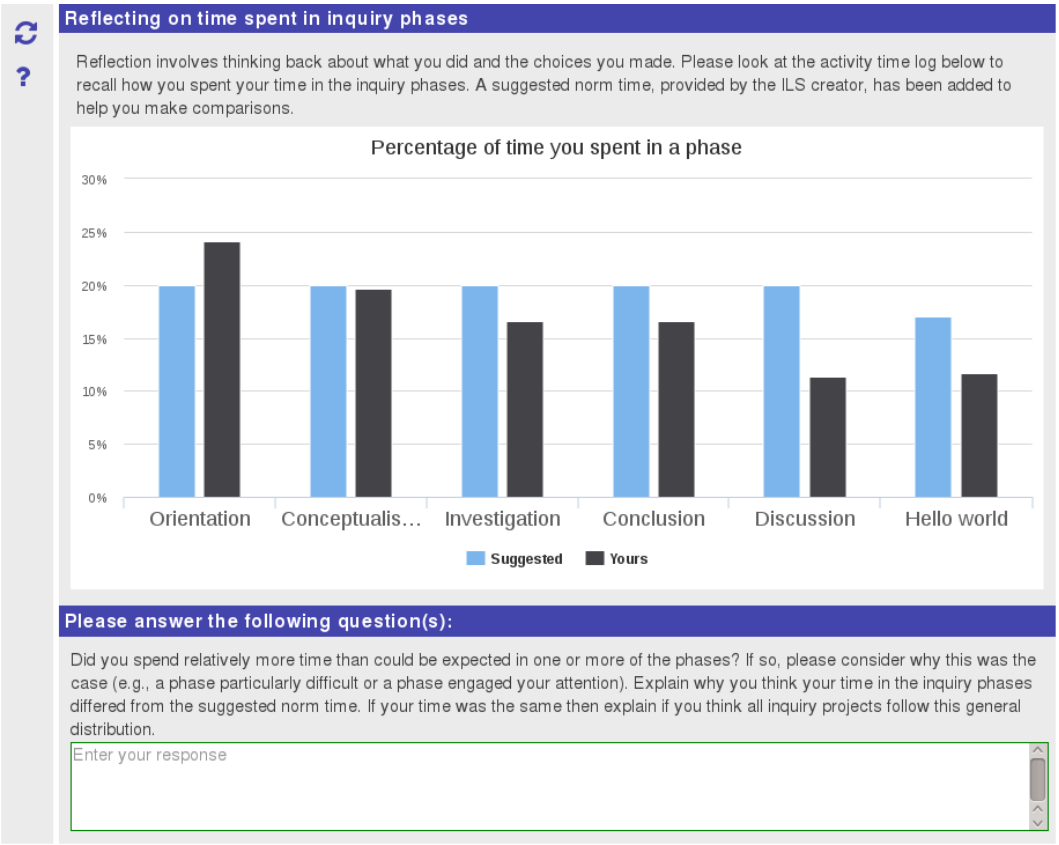


Figure 4. The sequence of inquiry phases visited by a learner.

for how long. Again, the teacher can define questions to prompt students to initiate the reflection process.

Both (client-based) reflection tools query a reflection agent running on the LA server. This agent monitors incoming actions and collects the time spent and phase transition actions for all students in all ILSs. The two reflection tools query the agent for the aggregated data for a particular student in an ILS by providing the studentId and ilsId as parameters. The agent sends the aggregated data to the tool which creates the chart visualisation shown to the student.

2.6.2 Example 2: My learning process

The app "My learning process" provides a visualization of the learners activity sequences, representing actions in phases and apps. The main usage is to present participation-based data to the learner, such as a sequence of actions or the time spent in phases. Figure 6 shows this app. The bigger circles represent the different inquiry phases, while the smaller ones, which are contained, stand for the different apps used in each phase. A connection between two apps will be established, if there is a consecutive action, while a connection between phases indicates the phase change. The size of the phases corresponds to the time that a student spent in each phase. Empty phase nodes stand for phases that have no tools inserted that is using action logging.

There are mainly two data sources that are used for the rendering of the visu-

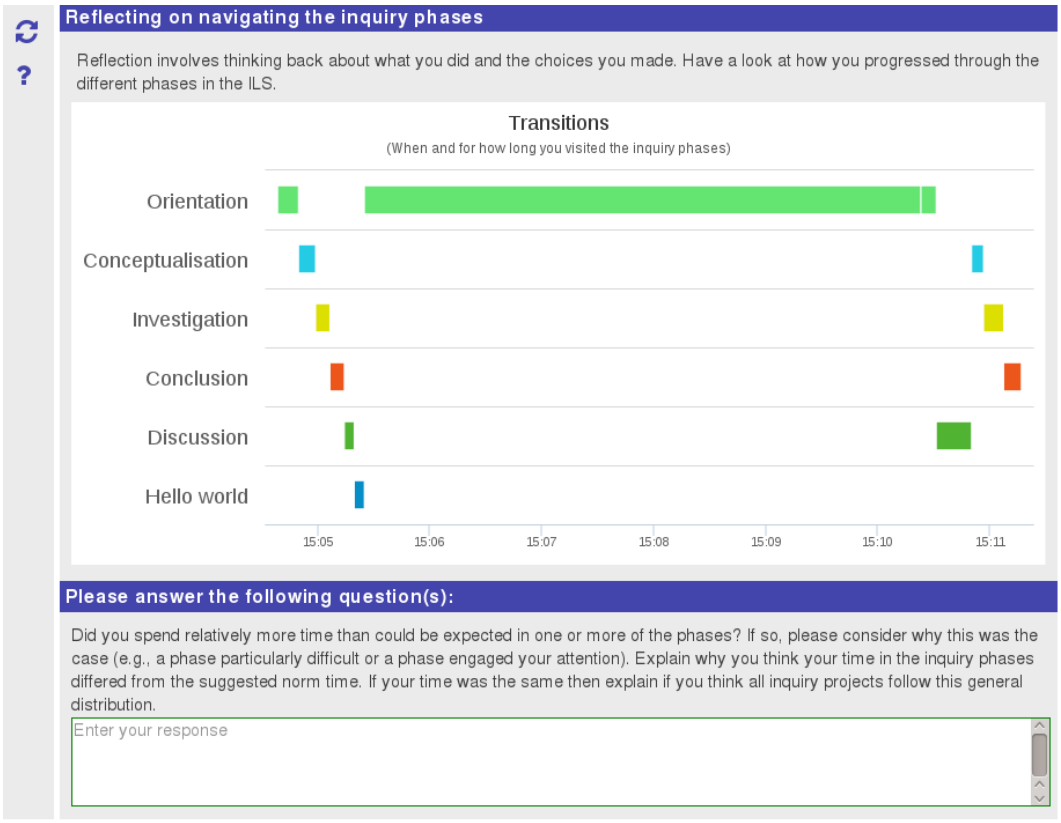


Figure 5. Time spent in inquiry phases: comparison of individual time spent to a teacher-induced norm.

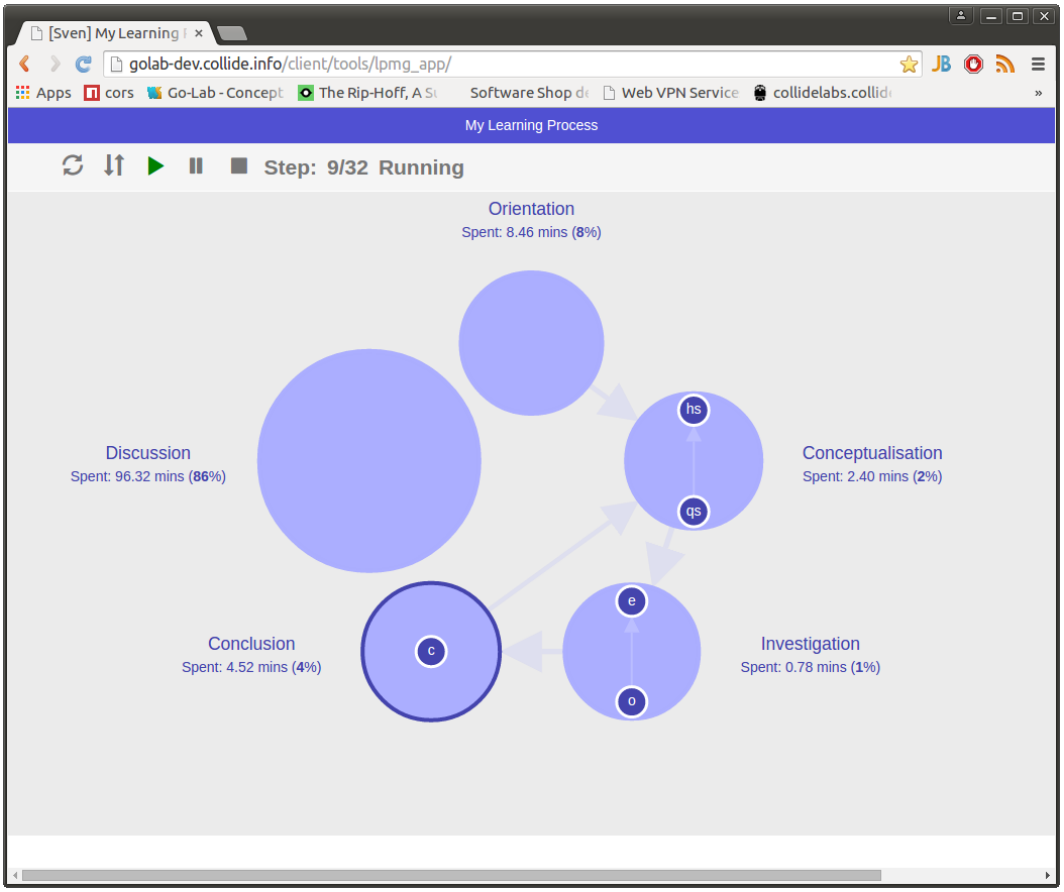


Figure 6. The app "My learning process" presents participation-based data to the learner.

alization: (1) action logs, and (2) contextual information. The action logs are processed on the server side, while the context is extracted client-side. The app retrieves the context, particularly the phases in the ILS where the app is running in, from the Go-Lab integration libraries. This is important on the one hand to ensure a consistent ordering independent from the action logs, and on the other hand to also display phases which did not cause any action logs, e.g. phases that have not been visited. The service for the retrieval of the data model is based on this context and (mainly) action logs. The action logs will be aggregated and processed so that a model of the sequences of phases and apps used is created. This model corresponds to the learning process on the part of the ILS platform. A detailed description of the implementation will be offered in D4.8, as well as a description of the service interfaces.

2.6.3 Example 3: Concept Map Aggregation

The "Concept Map Aggregation" app has been created through the analytics workbench and deployed as a micro-service inside this architecture, which has been mainly described in deliverable D4.4, section 2.4.7. Regarding the data sources, this also differs from the former examples as this is not presenting participation-oriented data based on action logs, but artefact-related information, namely students' concept maps. The service is retrieving the concept maps for an ILS, converting it to a common graph representation and merging the graphs based on node labels. The resulting aggregated graph is then visualized. The choice of a different layout compared to a concept map in Go-Lab has mainly two reasons: on the one hand, the layout is more compact and on the other hand, it makes clear that there are differences in the structure of a merged graph compared to a concept map. To reduce the amount of information and the possible information overload, edge labels are removed for the matching. Figure 7 shows an aggregated concept map that has been created during the implementation activities of Go-Lab. It combines 16 students' concept maps (in german) about osmotic power plants. The node and edge size of the visualization is dependent on the occurrences of a specific entity. In the figure, concepts such as turbine ("Turbine") or salt water ("Salzwasser") are more frequent than others (e.g. "Filter") and even more frequently connected.

2.7 Advanced LA apps for teachers and the teacher dashboard

In accordance with the previously assessed information needs of stakeholders (D4.2, section 2.3.1), teachers will be supported with monitoring and supervision facilities, realized through rich visualisations of different indicators. The data sources are mainly actions, but also artefacts for aggregative visualisations such as the "Concept Map Aggregation" 2.6.3 app. One of the challenges for such a dashboard will be, a minimum overlap or redundancy in terms of metrics or indicators to be visualised and an added value for the users. When visualising app-specific indicators, not every ILS will use a certain type of app leading to a possible loss of meaning in the dashboard as a whole. Therefore we propose a templating concept, which is based on several typical use cases, which have

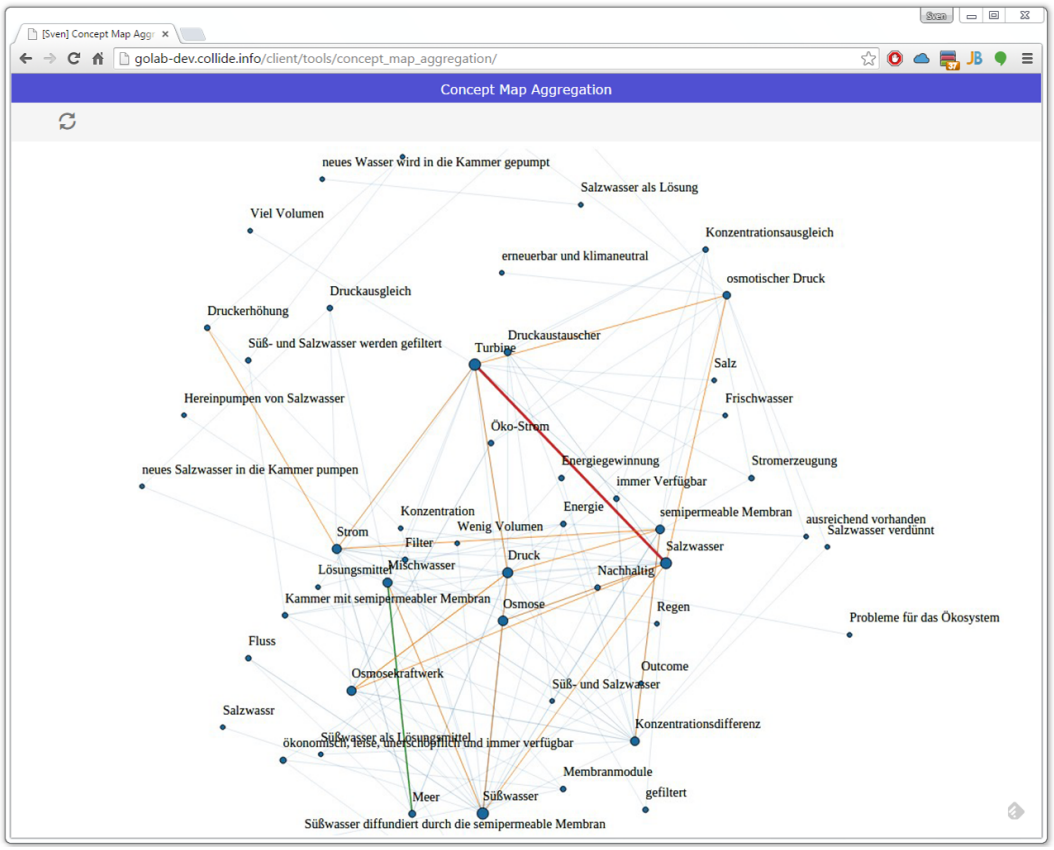


Figure 7. An aggregated concept map from a classroom scenario, merged from 16 student concept maps.

been identified in participatory design workshops and implementation activities.

2.7.1 Dashboard templates

On the technical side of the Graasp-Portal, a dashboard is implemented as a private subspace of an ILS. Therefore, it does not appear to students or non-members of the ILS and will be invisible for them. This makes it in line with the Go-Lab privacy policy and hides the potential personal information from externals. Similar to other ILSs or spaces, it will be a collection of apps. To enable access to the teacher dashboard, a templating mechanism similar to ILS templates will be used. Each dashboard template is a preconfigured space, which contains a set of dashboard apps. The teacher might want to adapt this dashboard and add or remove apps from the dashboard. Each template will suite a certain implementation scenario according to an ILS. Several apps exist that make only sense in a specific configuration, e.g., the "concept map aggregation" requires concept maps to be created first.

2.7.2 Indicators and apps

The dashboard consists of several apps and indicators. To reduce the risk of information overload, it is composed in a way that most apps do not present overlapping information. The composition of each template is based on ILSs that have been evaluated from the pedagogical cluster and identified as prototypical examples, as well as through participatory design activities. The apps contained in each dashboard template will cover both action- and artefact-oriented indicators, for example activity statistics, representations of learning processes or phase sequence, or to some extent apps to review artefacts of a certain type. The different types of compositions and dashboards will be presented in the release deliverable D4.8.

2.7.3 Example

The deliverable D4.8 for the release of the learning analytics services will describe the implemented learning analytics apps in detail as well as the different templates and preconfigurations of teacher dashboards. Some of the apps have been introduced in the initial release in deliverable D4.4. In this section we give a brief overview of one example dashboard with short descriptions of the used indicators. Figure 8 shows an example configuration of such a dashboard. It presents action- and artefact-oriented indicators to the teacher in an aggregated way. The dashboard consists of the following apps:

- Phase Timeline: students in phases,
- Action Statistics,
- Concept Map Aggregation,
- Students in phases over time.

The first app to be displayed ("Phase Timeline") is a monitoring app to supervise the sequence of phases that students visited over time. The right side of the chart canvas reflects the current phase a student is in. This visualization eas-

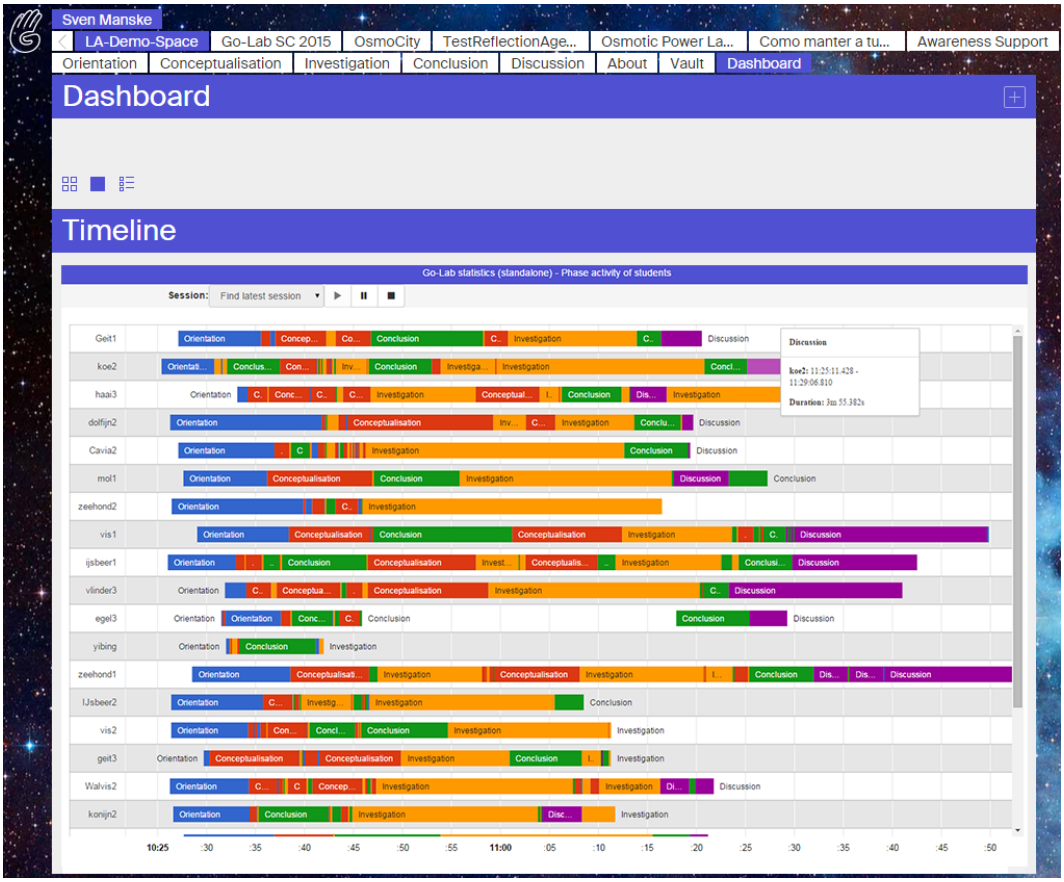


Figure 8. Dashboard example: the "Phase Timeline" is the first app displayed in this teacher dashboard.

ily uncovers out-of-order-behaviour of the students, such as jumping between phases or a very short time spent in a single phase. Such an app provides an overview of the activity in the whole class, but still providing easy access to actions and learning processes of individuals. The app will be described detailed in D4.8.

The second app is displaying "Action Statistics", particularly a visualization of the activity distribution across different apps used by different learners. Teachers can also use it ex-post to grasp approximately the amount of activity that the students spent. The "Concept Mag Aggregation" provides an overview about the aggregation of all students' concept maps. By using this, a teacher might discover misconception or increase the awareness for important and not used or non frequent core concepts for the learning scenario. This app is an example of an LA app that can be used both for learners and teachers. These apps have been already described in the deliverables D4.4 and D5.3.

2.8 User evaluation of initial versions of apps for teachers and students

This section reports on the first evaluations carried out at the GoLab summer school organized in Athens in July 2015. The participants were 27 teachers who

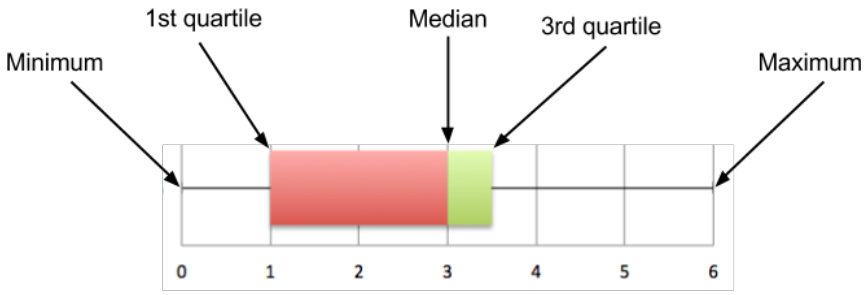


Figure 9. Reading a Box-and-Whishker Plot.

had already used ILSs in their own courses. Thus, they may provide a valuable feedback about the LA apps.

During the workshop, the teachers were distributed in groups of 4 and each group evaluated the apps according to the following aspects using a 7 point Likert scale:

- General evaluation of the app: bad (0) - good (6)
- Design of the the app in terms of user interface: ugly (0) - attractive (6)
- Usefulness of the app: not useful (0) - very useful (6)
- Presentation of the information: confusing (0) - clearly structured (6)

Additionally, the teachers were encouraged to provide comments and recommendations that could help us to improve the apps.

The following subsections are organized according to the target users. I.e., on the one hand we present the outcomes of the LA apps created for the students and, afterwords, the ones devoted to the teachers. In order to present the data collected, the following graphs provide the box and whiskers plots, also known as five statistical summary or five number summary. Looking at Figure 9, in these charts we can identify: the minimum (the least score received), the first quartile (i.e., 25% of data less than this value), the median (50% of data is greater than this value), the third quartile (25% of data greater than this value) and the maximum (the greatest score received). Such information shows the distribution of the data set at a glance, revealing the extent to which the data is located near the median or near the extremes.

2.8.1 Evaluation of LA apps for students

Time Spent in Phases. As it is shown in Figure 10, this app received a high acceptance from teachers. In general terms it was considered a good app (with an average of 5), useful, with an attractive interface and clearly structured. The only drawback identified was that suggesting an ideal amount of time to the different phases is not conceptually easy.

My Learning Process. The feedback received about this app varied (see Figure 11). The average was 3, but half of the teachers rated it positively and the other half had an opposite opinion. While the user interface was attractive for

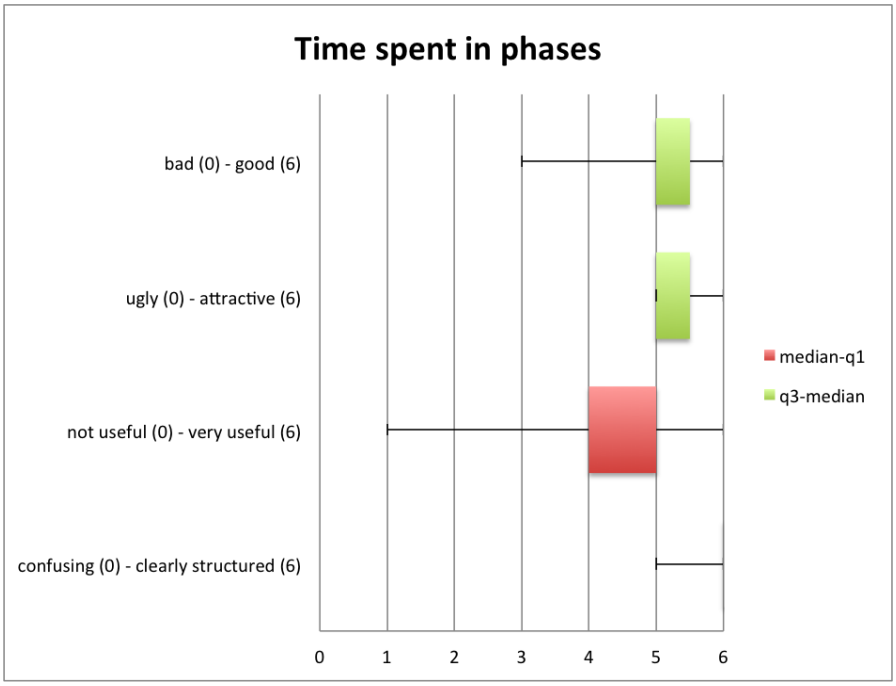


Figure 10. Evaluation of the app: Time Spent in Phases.

the teachers, the main problems were related to the usefulness and the information offered. The teachers reported that it would be necessary to provide an explanation of the diagram and its implications since it is not evident how the app may help the students. This might be caused by the difficulty to interpret such representations and the necessity to engage learners actively with it. As a potential improvement, the representation can be complemented by adding questions targeting the learners to reflect about specific singularities or deviations from the recommended inquiry model, such as loops in the visited phases.

Concept Map Aggregation. Figure 12 shows that the app received a positive evaluation from the teachers (average of 4.57). Regarding the way the information is presented, the teachers pointed out that in case of having many students, the visualization could be ugly. In terms of usability, a couple of groups had problems in order to know how the app works (e.g., how many maps are necessary). Additionally, two groups highlighted that they did not understand the purpose of the app by themselves, and they required an explanation from the workshop organizers; and a third group considered that the analysis offered by the app is very difficult for the students.

2.8.2 Evaluation of LA apps for teachers

Action Statistics. The app obtained an average of 3.29 points. Figure 13 shows that there were both supporters and detractors. First of all, there was a common misunderstanding among participants regarding what the the app measures (the y-axis is not defined) and what kind of data is used from the different apps. In terms of visualization, the participants argued that the app could be confusing in case they had many students. Moreover, they complained

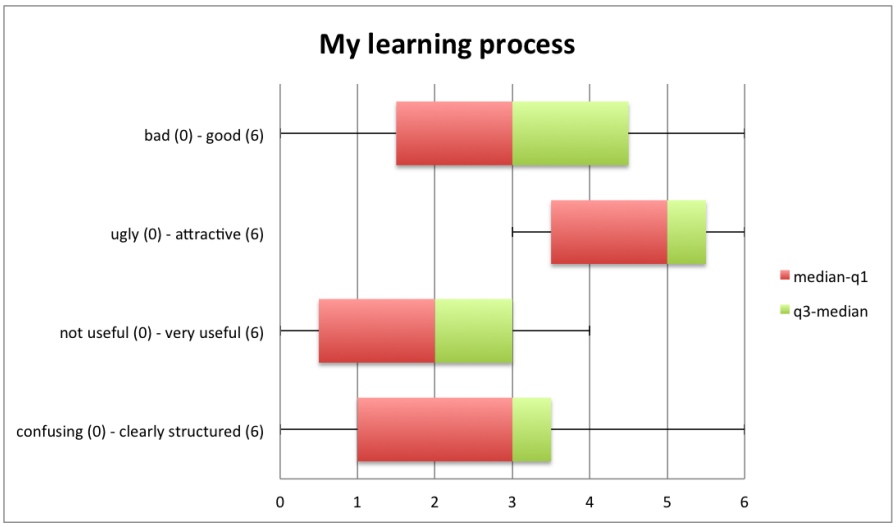


Figure 11. Evaluation of the app: My Learning Process.

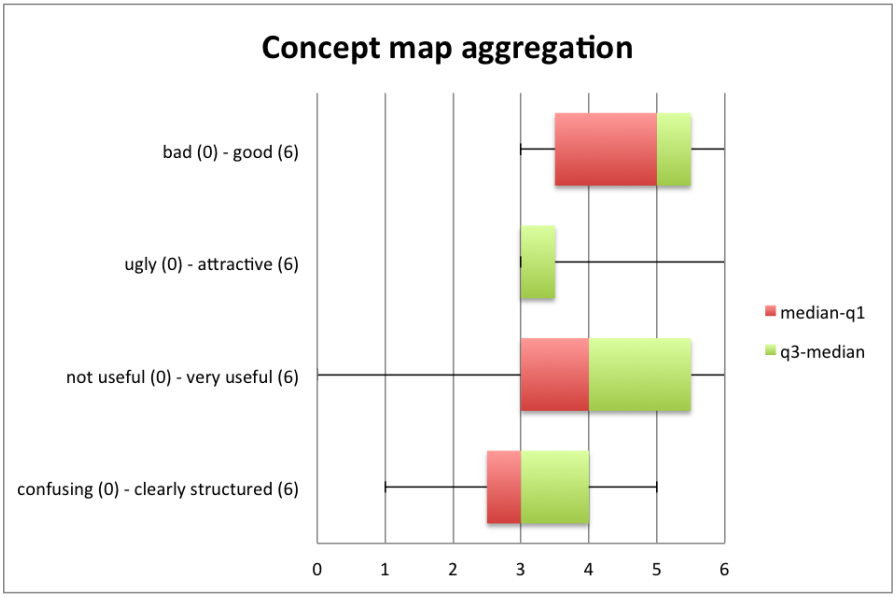


Figure 12. Evaluation of the app: Concept Map Aggregation



Figure 13. Evaluation of the app: Action Statistics.

about the amount of information presented at the same time. However, it was caused because they did not realize about the possibility of showing/hiding the information retrieved from the different sources.

Active Students. The average rating given by teachers to this app was 5 points. As it is shown in Figure 14, the user interface, the usefulness and the clarity of the information presented were positively evaluated. The qualitative feedback rises one relevant issue about the usefulness of the awareness apps. As the participants mentioned, the usefulness depends on the teaching style: if the teacher usually walks around the class instead of staying next to the computer, she will not be able to pay attention to these type of apps.

Number of Students per Phase. The app received an average of 2.71 scores. Figure 15 reveals that the main problems were related to the usefulness and the visualization of the information. The teachers mentioned in their comments that the graphs should be improved in order to increase the readability and understanding. For instance, they suggested to enlarge the time periods so that the fluctuations are smaller, and to include the option of selecting only one phase to be displayed.

Submitted Reports App. As it is shown in Figure 16, this app received a high acceptance from teachers (average 5.57), with no explicit complaint in relation to the design, usefulness and data visualization. Several ideas emerged from the teachers in order to improve the tool, namely: group the files by user and allowing the option to click and see a preview.

Time Spent. Figure 17 presents the results of the evaluation. The app was rated by the teachers with 5.57 scores (average). Despite considering the app useful, the teachers had doubts about how the app was measuring the time spent. They noticed that, even if the app stops adding time after certain amount

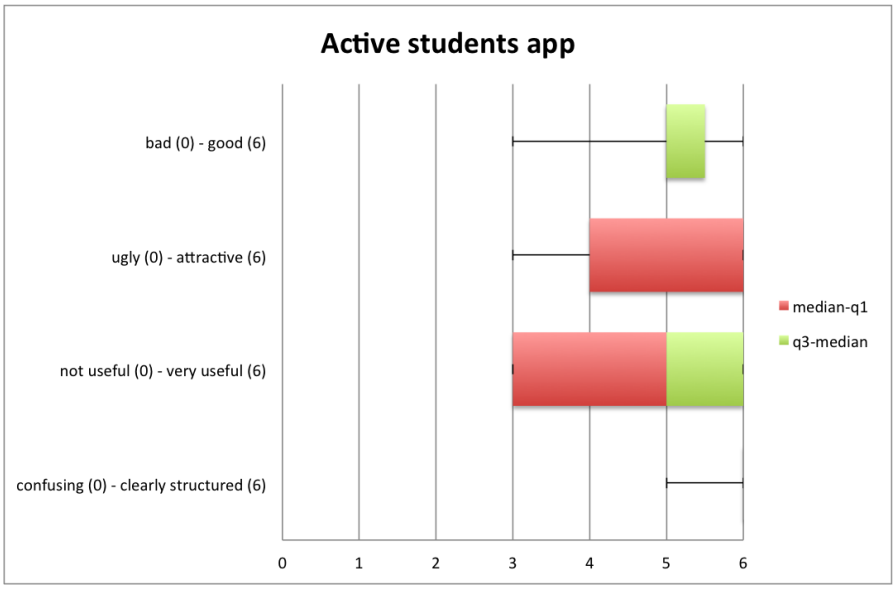


Figure 14. Evaluation of the app: Active Students.

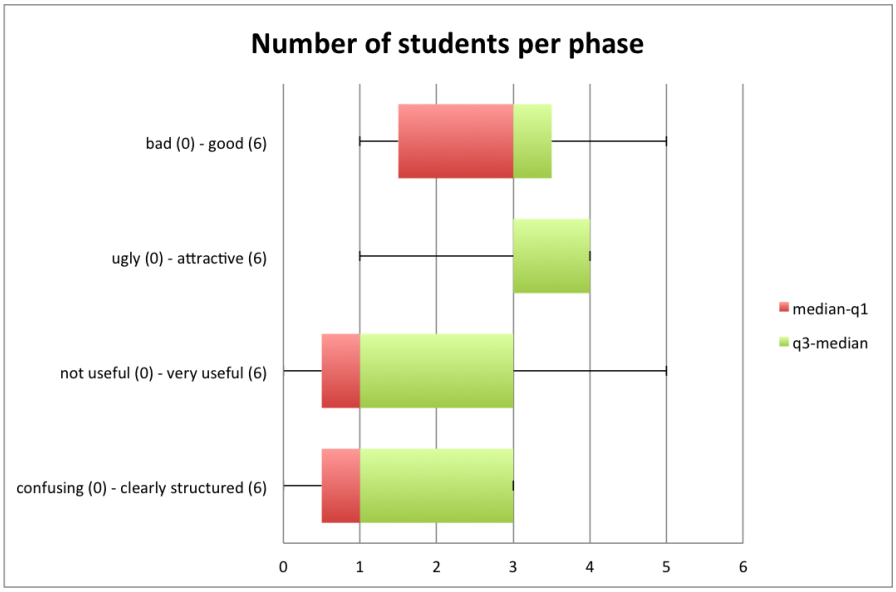


Figure 15. Evaluation of the app: Number of Students per Phase.

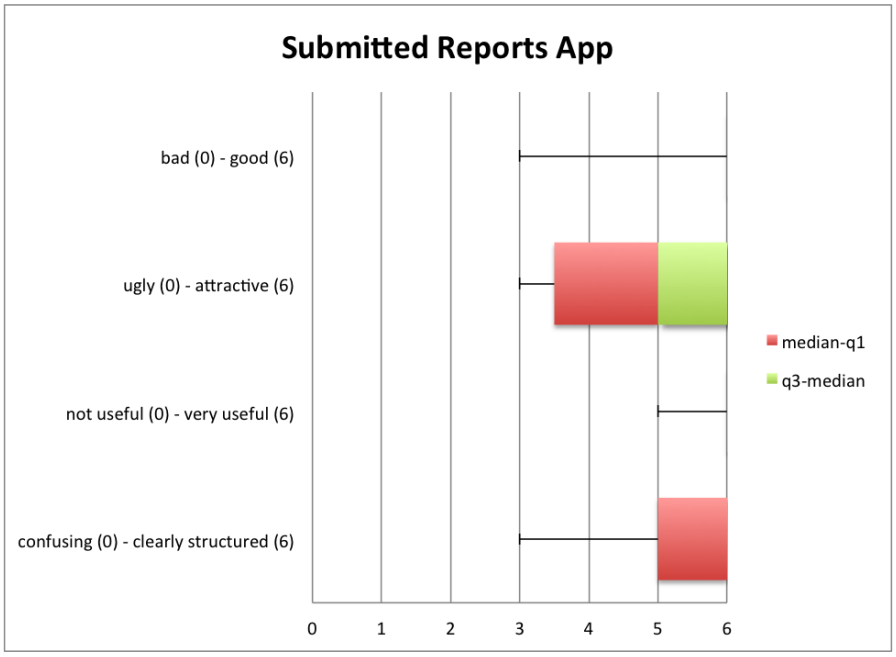


Figure 16. Evaluation of the app: Submitted Reports App

of time without receiving any action to be logged, this indicator may be imprecise (e.g., the student is just reading or looking at a video without interacting with the ILS). As potential improvements, the teachers came up with some ideas for the user interface (e.g. fixing the headings of the columns when scrolling) and regarding the potential benefits of sharing the app with the students (for self and peer-evaluation)

2.9 Specification of the Golabz recommender system

The resources on the Go-Lab Repository (see D5.2, and available at `http://golabz.eu`) have been growing fast, e.g. the online labs have almost doubled in the last year (as of July 2015, 145 labs are available) and teachers are sharing more and more inquiry learning spaces. To support the user in finding content on the repository and enabling the discovery of new content, a recommender system has been developed. In D5.1, we have described the preliminary specifications for the recommender system. In the meantime, the needs for such a recommender system have changed and based on the review recommendations, we have redesigned the specifications for this recommender system (originally described in D5.1) to be more in-line with the needs of the project and its users, while limiting the complexity of its implementation.

In this section, we elaborate on the design of the recommender system and the design choices made, as well as discuss the technology choices made.

2.9.1 Data sources

To create recommendations of items in the Go-Lab Repository, different data sources are available that can be of use. All online labs, apps and ILS in the Go-Lab Repository are described by metadata (see D2.2). All pages of the Go-

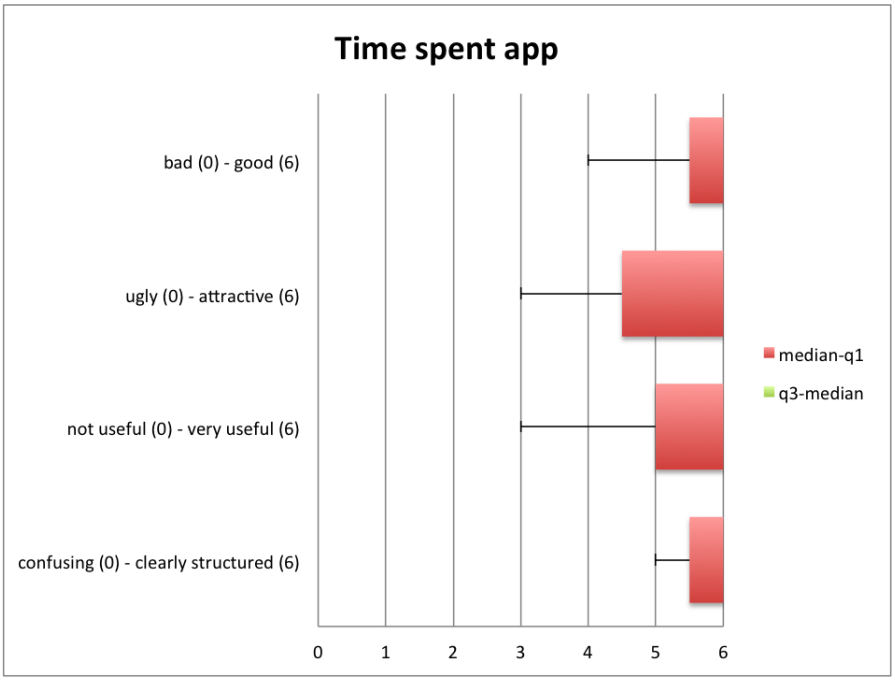


Figure 17. Evaluation of the app: Time Spent.

Lab Repository are tracked using Piwik[4], and Google Analytics[5], and Drupal Statistics to collect page views anonymously. Furthermore, there is also the structure of an ILS, which can show which apps and online labs are used in combination.

2.9.2 Design of the recommender system

The recommender system is integrated in the detail pages of online labs, apps and inquiry spaces on the Go-Lab Repository. This enables teachers and visitors to discover related resources on the detail pages and directly navigate to the recommendations from these pages. This is the main scenario. A secondary scenario could be that a teacher is creating an ILS on the ILS Platform (Graasp). While editing the ILS, a list of potential other apps and labs that can be added to her ILS is shown in a separate app or directly in Graasp.

Due to the type of available data sources, the recommender system is designed as a hybrid recommender, focusing on content-based filtering making use of the Go-Lab Repository metadata and the structure of ILS, combined with the usage statistics of the anonymous page views. Below, we elaborate on which metadata fields are used to create recommendations for apps and online labs.

2.9.2.1 Recommendations for apps

Apps do not have an extensive set of metadata fields available. Therefore, the content of different metadata fields is taken together and based on this text, similar apps are retrieved using fuzzy search algorithms. We focus the recom-

[4]Piwik, `http://piwik.org`
[5]Google Analytics, `http://analytics.google.com`

mendations on the following fields, when available:

- *Title*: The title can contain words that reoccur in the metadata fields of other items.
- *Description*: The description is one of the richest metadata sources for apps.
- *Category*: There is a list of six app categories available (i.e. 'Go-Lab inquiry apps', 'General apps', 'Collaboration apps', 'Learning analytics apps', 'Math related support apps', & 'Domain specific apps'). These categories represent a simple grouping among apps, however apps in different categories can still be good recommendations.
- *Keywords*: Keywords are added to further classify apps, but they are used as open folksonomies, so it is hard to only filter on these terms, that is why they are combined with the other metadata fields.
- *Views*: The page views are used to order the recommendations.

2.9.2.2 Recommendations for online labs

Since the online lab metadata has a much richer metadata field set (see D2.2), the recommender system is a bit more complex. Below all metadata fields are listed which are used when available:

- *Title*: The title can contain words that reoccur in the metadata fields of other items.
- *Description*: The description is also used as it can contain extra information not related to other metadata fields.
- *Grade level*: The grade level is useful to focus the lab recommendations to a student target age. The recommendations should have the same grade level to make sure they target the same student age.
- *Language*: Similarly to grade level, a good recommended lab should support the same UI language. However, it is not limiting the recommendations to all have the same UI language.
- *The Big Ideas of Science*: The Big Ideas organize the labs in 8 categories. These categories provide a rough relationship between labs.
- *Subject domains*: Each lab is described with a fine-grained taxonomy of subject domains, which provide a good way to find similar labs.
- *Keywords*: As for apps, labs are also described using keywords which can be used as a folksonomy for recommendations.
- *Lab owner*: Often a lab owner makes quite similar labs, e.g. focusing on a specific subject like biology.
- *Views*: The page views are used to order the recommendations.

Only the grade level is required and every recommendation should share the same grade level to ensure the recommended labs target the same students.

The other fields are used to filter related items. When more fields are available and contain similar content, the similarity will increase. The same as with apps, fuzzy search algorithms are used to determine this similarity.

2.9.3 Technology specification

As mentioned the recommender system is used in the Go-Lab Repository, but could also be used by the ILS Platform. Therefore, we have opted to implement the recommender system as a separate service from both platforms. Figure 18 illustrates the architecture.

The Go-Lab Repository is implemented on top of Drupal as described in D5.2 and uses a MySQL server to store its data (see 1 in Figure 18). To support the fuzzy text search and other text retrieval algorithms, the recommender needs a more advanced data structure. We have opted to use ElasticSearch[6], an open-source search engine. ElasticSearch offers various text-based similarity and fuzzy search options. Furthermore, ElasticSearch is highly efficient and can run distributed over many clusters, thus ensuring real-time execution of the recommendations. However, with the introduction of ElasticSearch, the Go-Lab Repository metadata needs to be imported and synchronised between the MySQL database used by Drupal and ElasticSearch to ensure up-to-date recommendations. Several synchronisation modules between Drupal and ElasticSearch exist (see 2 & 3 in Fig. 18). The recommender itself is implemented as a separate service, making use of the ElasticSearch search engine. To integrate the recommendations in Drupal a Drupal module is developed that sends a request to the recommender service (see 4 & 5 in Fig. 18) and displays the recommendations on the detail pages. The exact implementation details will be described in deliverable D4.8 and D5.6. In this D4.8 deliverable, we will also elaborate further on the recommendations for ILS.

[6] ElasticSearch, `http://www.elastic.co`

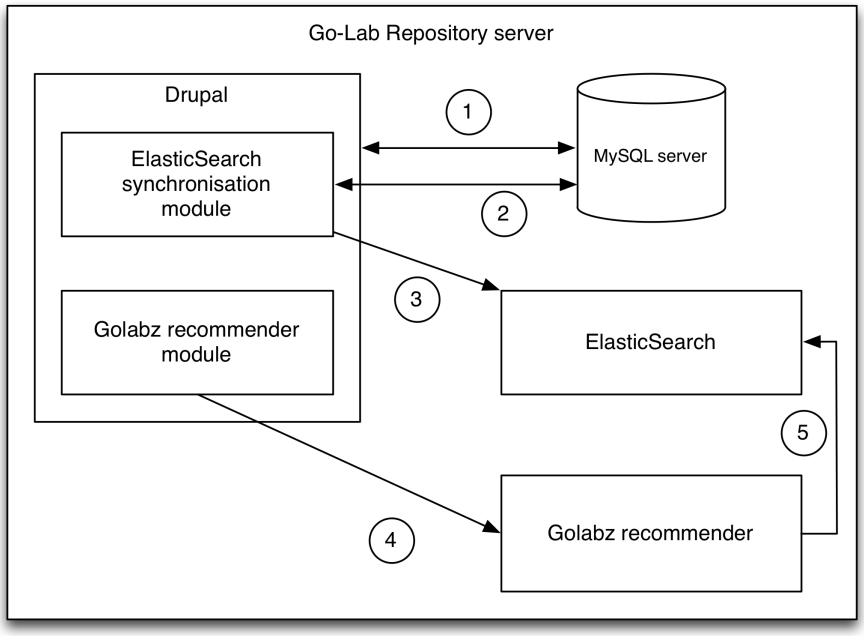


Figure 18. Architecture of the Go-Labz recommender system.

3 Add-on services

3.1 Introduction to Go-Lab add-on services

Besides the learning analytics as Go-Lab infrastructure services, two add-on services are specified to give lab owner added-value services and to support Go-Lab teachers community. This section is structured in two parts: the Booking System for remote labs and the Tutoring Platform. The “Tutoring Platform” is the new name and also an extension of the former “ Bartering Platform” used in D4.2. Based on the feedback from core teacher groups and in order to support teachers better, we have renamed the platform from the bartering platform to the tutoring platform since the release of the deliverable D4.4.

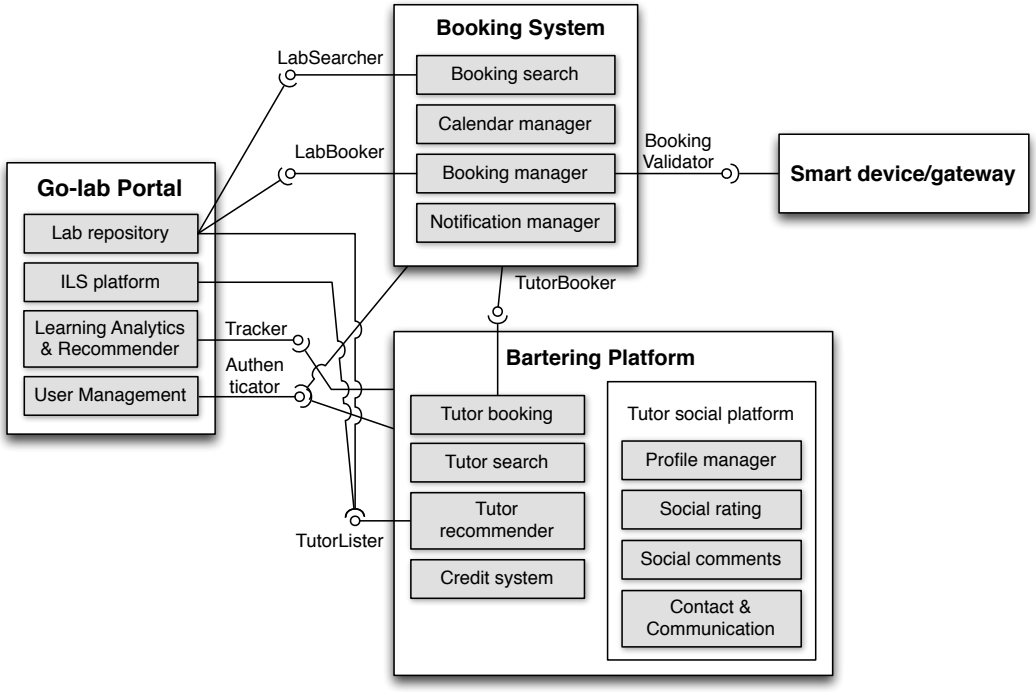


Figure 19. Architecture of the Go-Lab Add-on Services.

For an overview, the architecture of the add-on services is depicted in Figure 19. The Booking System offers the booking services explicitly to the scope of remote labs which often have limited resources. On the contrast, virtual labs could be used by a large audience online at the same time. Therefore, in the booking system, we don’t consider virtual labs. To complete the final specifications in D4.6, we have talked with (remote) lab owners and asked for feedback. According to lab owners’ feedback, the requirements and use scenarios have been rarely refined or revised since the initial specifications. In this deliverable, we focus on the integration implementation plan of the Booking System into the Go-Lab Portal and the Smart Gateway. Figure 20 shows the user interface of the lab calendars. A Go-Lab lab owner could create a lab use sessions for his/her remote lab and manage the calendar. A teacher could book a lab session using

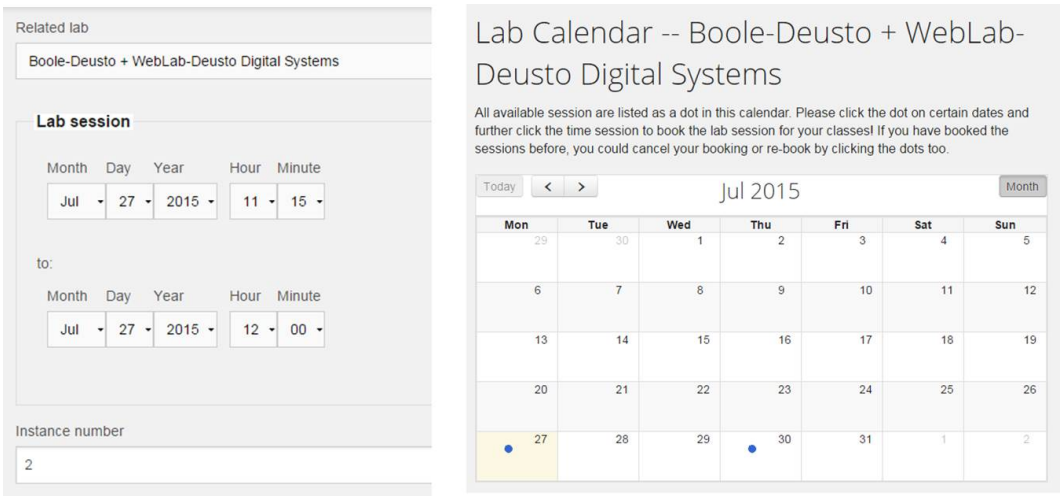


Figure 20. User interface of the lab calendar of the Go-Lab Booking System in Golabz.

the lab calendar. Teachers and remote labs may have different time zones in Europe, dealing with different time zones is a new feature of the booking system.

The initial release of the Go-Lab Tutoring Platform is accessible at `http://tutoring.golabz.eu/`. Some initial activities have been organised to use the Go-Lab Tutoring Platform. The first feedback is positive and constructive. Considering the feedback from the Go-Lab user communities, we refine the use scenarios and requirements in this section. We illustrate the extended use scenario in Figure 21: the Go-Lab Tutoring Platform support teachers' community in developing their Go-Lab knowledge and skills.

In addition, a bachelor thesis on this tutoring platform has been successfully conducted at IMC AG. The media informatics student Faysal Cherradi finished his thesis on the tutoring platform as a development support to Go-Lab Consortium (Cherradi, 2015). Some survey studies have been conducted and the feedback contributes to refine the final specifications too.

Based on the feedback both from the Go-Lab user communities and from the evaluation of the bachelor thesis, main changes include the following features.

- Broadcasting and Recording the tutoring sessions as Webinars;
- A forum for tutors and users as a discussion tool for community building.
- Notification and booking features consider different time zones.

More details are described later in this chapter.

3.2 Go-Lab Booking System

The Go-Lab Booking System for remote labs connects to the front-end lab booking user interface in the Go-Lab Portal (cf. Figure 20), managed in the Go-Lab Repository. The back-end is supported by the Smart Gateway where remote labs are hosted. The Go-Lab remote labs are set up as Smart Devices to con-

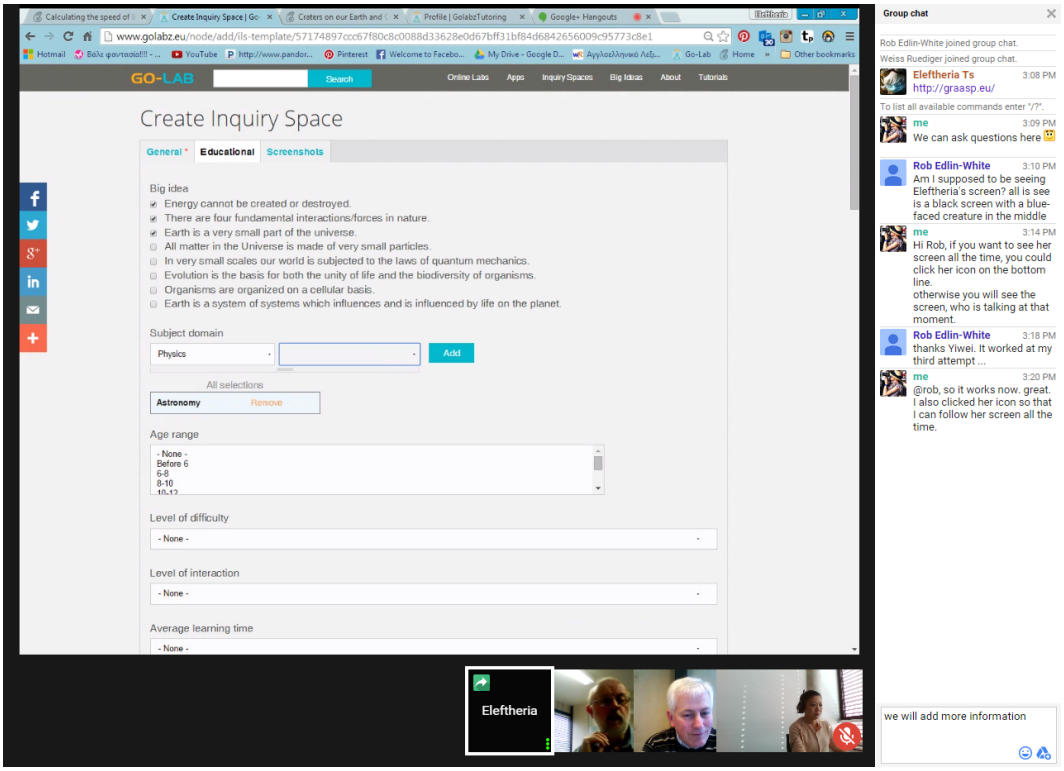


Figure 21. Teachers are shown how to use the Go-Lab Portal in the tutoring platform.

nect to Smart Gateway. The Smart Gateway provides a unified interface to enable external laboratories to be integrated in Go-Lab. It supports a set of plug-ins for different laboratory management systems. Each plug-in may support one or more laboratories. The user access grant mechanism used in the Smart Gateway is based on token provision and reservation. When a remote lab instance is called, the Smart Gateway validates the users and their booking information through the Go-Lab Booking System. If the information is valid, the Smart Gateway grant the users' access to the certain remote lab.

3.2.1 Requirements for the Go-Lab booking system

Based on the use scenarios and lab owners' reaction, the requirements are refined as follows. The changes are the two kinds of functionality *Search ready-to-book labs* and *Notifying lab booking progress in ILS* are graded to optional features (nice-to-have).

- **Single sign-on.** The add-on services need a simple authentication mechanism in line with the Go-Lab Portal. Go-Lab Inquiry Learning Platform (Graasp) user login and management is also used for the booking system.
- **Booking a lab.** To provide a consistent user experience, the complexity and diversity of the different booking systems of remote labs should have a unified user interface for teachers to book a lab. Since Go-Lab teachers work in different European countries, the lab facility and the teacher and its class could be in different time zones. So the booking system supports

different time zones.

- **Consulting lab calendar.** Teachers can access the calendar with lab availability and make a reservation of a certain remote lab.
- **Administer lab calendar.** Lab owners can add and edit time slots in the lab calendar to show when their labs will be available to Go-Lab users. Lab owners can indicate how many physical instances they provide access to. For instance, there will be many physical copies of the RED lab[1] available that could be booked separately.
- **Booking for multiple instances.** Teachers can input the number of lab instances when booking, if multiple physical instances of this remote lab are appropriate for students' use at school.
- **Running a booked lab in an ILS.** Students and teachers need to be able to effortlessly execute a booked remote lab in their ILS. Booking information and data communication of the experiment and its progress has to pass through the ILS. Teachers could either give the information of the ILS in use during their booking, or gets a new ILS if no pre-specified ILS is available.
- **Notifying booking.** Teachers need to be notified via email or in the Go-Lab portal about their booking at the time of the booking and a few hours before the booking for awareness cues. It considers different time zones as well.
- Notifying lab booking progress in ILS (nice-to-have). Students and teachers need to be provided with time left and total time information of the booking in the lab client app in the ILS. If didactic hours are considered, both teachers and students are aware of the lab usage time. Thus, this function could get a lower priority for implementation.
- **Cancelling booking.** Teachers need to be able to cancel their lab booking to deal with mistakes. This function may take place in two circumstances. During the lab booking process, teachers can cancel the booking at any step before they confirm the booking. During the time period between a successful booking and the lab usage time, teachers are able to cancel their bookings, if they don't need the labs later. The occupation status of the labs will be changed for future re-booking by other teachers.
- **Closing a booked lab session.** Once the reservation time has expired, the remote lab experiment has to be gracefully ended in the ILS. The lab could for instance only allow observation and no longer operation.
- Searching ready-to-book labs (nice-to-have). This functionality is downgraded as an optional one. The reason is that each lab is special and unique with the experiences of lab registration in the portal. Thus, it is useful to support teachers in searching another lab if the one he looks for is not available.

[1] RED lab, `http://www.golabz.eu/lab/red-lab`

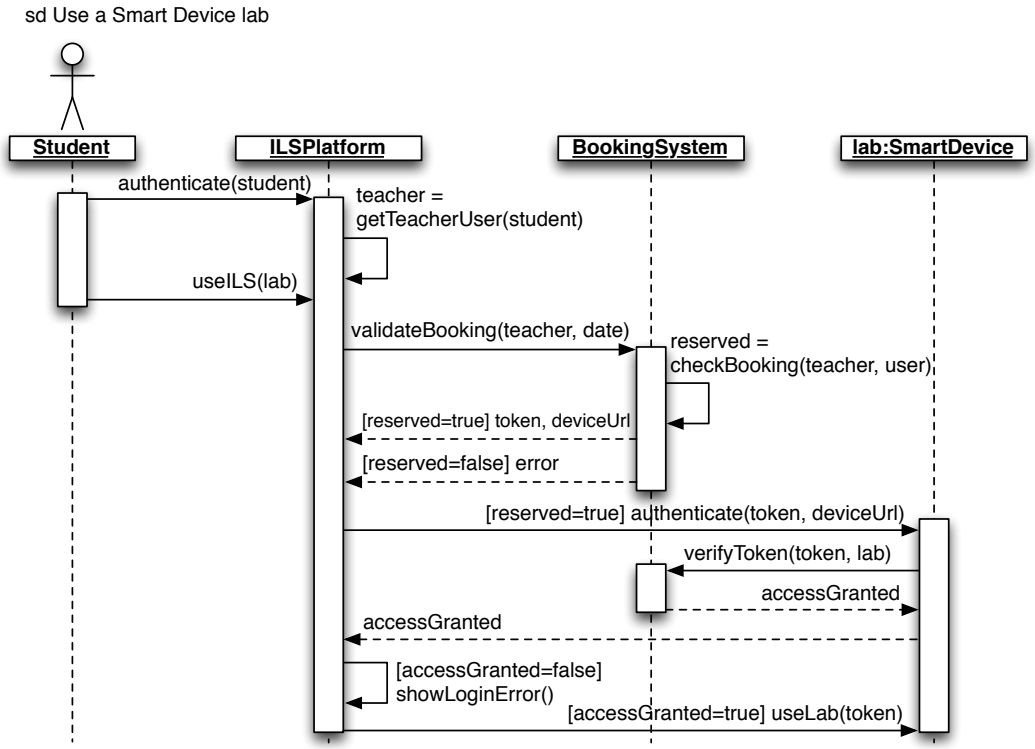


Figure 22. A sequence diagram that models how a booked lab using a smart device is used in an ILS.

- **Booking statistics**. Web statistics will be tracked, including how many users have clicked the booking buttons and how many bookings have been made. This is an added-value service for lab owners to obtain information and overview of their own remote labs.
- **Booking of other resources than labs.** The booking system can also be used to book lab tutor time and other scarce resources. The tutoring platform use the same booking system as depicted in the architecture diagram (cf. Figure 19).

3.2.2 Integration specification of the Go-Lab Booking System

The booking system works together with Go-Lab Repository, ILS Platform (Graasp), Smart Gateway, and Smart Devices, as specified in the sequence diagrams in the initial specifications D4.2.

Additionally, we consider the scenario that the booking system should not block a Go-Lab remote lab if no users have booked it. A null message will be returned if no reservation has been made, and therefore the Smart Gateway will enable any user to access the lab at that time without booking.

We take the diagram of using a booked smart device lab as an example (see Figure 22). For both ILS Platform of the Go-Lab Portal and Smart devices, a plugin will be implemented to communicate with the Go-Lab Booking System via Web Services.

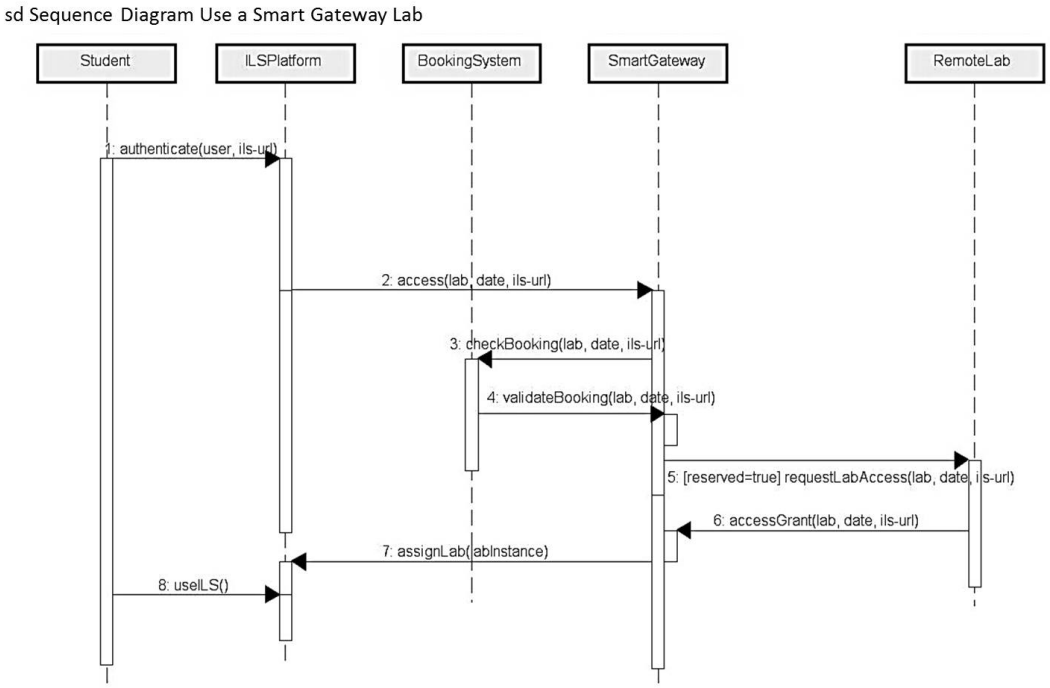


Figure 23. A sequence diagram that models how a booked remote lab hosted on the Smart Gateway is used in an ILS.

In the final specifications, the sequence diagram of using a Smart Gateway lab is added to cover all use cases in the booking systems completely (cf. all sequence diagrams in D4.2).

Figure 23 depicts the following use case. During a lab class, the students have the access of their ILS link shared by their teachers. They login in the ILS (Msg. 1) to access the ILS and its remote lab in the investigation phase of the inquiry learning spaces (Msg. 2). The Smart Gateway uses its plugin to exchange information with the Go-Lab Booking System (Msg. 3-4) to reject or grant students' access (Msg. 5-8).

The information exchange between the Smart Gateway and the Go-Lab Booking System is specified in JSON. The following code snippet gives an example of the JSON file.

```
{
        "id": "http:\/\/www.golabz.eu\/lab\/red-lab",
      "ils":"http:\/\/graasp.eu\/ils\/559edb60b5a072ca55
      6736e3\/?lang=en"
        "name": "RED Lab",
        "labWidgetUrls": [
                "http:\/\/128.178.5.64:8080/GL.xml"
        ],
        "user": "mabaced@gmail.com",
        "bookingInfo": {
                "beginTime": "2015-09-23T16:00:00.511Z",
```

```
                "endTime": "2015-09-23T17:00:00.511Z",
                "modifiedTime": "2015-07-15T09:25:43.
                                 511Z",
                "createdTime": "",
                "nbInstances": "1",
        }
}
```

Information about lab and its widget url, ILS url (the url for the student view), as well as the time slot is included in the JSON message between the booking system and the Smart Gateway and Smart Devices.

As we mentioned at the beginning of D4.6, it focuses on the integration specification of the Go-Lab Booking System. This specification grants a smooth final release of the booking system in cooperation with the three main parts: 1) Go-Lab Portal (repository, ILS platform), 2) Smart Gateway as well as Smart Devices, and 3) A number of remote labs.

3.3 Go-Lab Tutoring Platform

Teachers' ICT skills are limited and need further improving (Breuer, Klamma, Cao, & Vuorikari, 2009). Beyond ICT skills, it is vital to offer science (STEM) teachers professional development of pedagogical methods such as inquiry learning approaches and learning materials authoring. Jimoyiannis (Jimoyiannis, 2010) has designed a Technological Pedagogical Science Knowledge (TPACK) framework to cover the aspects to enhance teachers' competence, which is also used in Go-Lab Work Package 8. Thus, the Go-Lab Tutoring Platform could meet the requirements to offer teachers a platform for their professional development. Furthermore, it aims to build up an active teachers' community for vocational training and life-long learning (Cao, Govaerts, Dikke, Faltin, & Gillet, 2014).

The initial specification of the Go-Lab Tutoring Platform (D4.2) has been implemented, reaching the user communities. The initial release of the Go-Lab Tutoring Platform is `http://tutoring.golabz.eu` for testing and users' feedback. Review comments from the EC project reviewers focus on further employment of the platform in a wide user community and how the platform can support community building. To meet with the requirements, we have extended the use scenario for teachers vocational training and life-long learning. And the requirements are refined as well.

The original review is documented as follows as a reference:

> D4.2 - The bartering platform and the credit systems are key to Go-Lab sustainability. There is not yet enough evidence of the integration of its design with the pedagogical work package and the dissemination and exploitation work package, although there are signs of the awareness of its need (well recapped p.60). The management must take care that this cooperation does come too late implying adaptation costs which could have been avoided; the integration

of these crucial work packages should be precisely documented in the future activity reports. The initial recruitment of experts is important in the bootstrap period, since associating teachers with sufficient knowledge and skills using online labs and inquiry spaces will take time and depend on the effective deployment of Go-Lab.

In the aforementioned point of integration with dissemination and exploitation work package, we consider the new requirements of adding a forum into the tutoring platform to support the teachers' community. Moreover, measures have been carried out to promote teachers to use the tutoring platform. If teachers meet with problems in using online labs, learning apps, and inquiry learning spaces, they are encouraged to use the tutoring platform to meet with other teachers and Go-Lab experts.

In the aforementioned point of integration with pedagogical work package, the tutoring platform has been inviting experts in the research area of inquiry based learning (mostly the partners from WP 1) to join in the tutoring platform and offer tutoring sessions to teachers. Some first results of these measurement will be shortly introduced later in this section. More details of roll out of the Go-Lab Tutoring Platform will be reported in D4.8 the final release of the Go-Lab Add-On Services.

The review will be briefed at the end of this deliverable as a clear guide to the upcoming final release of the Go-Lab Tutoring Platform in D4.8.

3.3.1 Use scenario of the Go-Lab Tutoring Platform

The following use scenario of the initial release is added to the initial specifications (D4.2). The tutoring platform is employed to teachers' vocational training. Teachers and tutors will have user profiles to describe their expertise and their skills. Tutors offer training events for teachers. Tutors' reputation can evolve based on social ranking and commenting by teachers. Teachers obtain expertise from peer assistance. Thus, the ranking of the tutors may influence teachers' opinion on selecting a tutor's help. For example, tutors' help session will not be booked by teachers if the tutors receive poor ratings.

3.3.2 Requirements of the Go-Lab Tutoring Platform

The requirements are reviewed and listed as follows. The main change is the credits mechanism need be adapted to the new use cases, teachers' training events. They are based on feedback from WP2 and WP6.

- **Sign-on.** The tutoring platform as a Go-lab add-on service should be independent to the Go-Lab Portal. To achieve the goal of a user friendly login interface, same login information in the social platform user accounts such as Google or Facebook should be supported. Authentication is needed for most functionality of the tutoring platform (e.g., not for searching).
- **Managing a tutor profile.** A tutor can create a profile and update it according to tutors' help offers. We will show a badge if the tutor has given a session in the past. This helps users know more about the tutor.

- **Managing a user profile.** This is refined to be in line with the function of *managing a tutor profile*. Any user is a potential tutor in the tutoring platform.
- **Commenting and rating.** Users could comment and rate the tutors after they get help and support from the tutors. This will be reflected in the tutors' profiles. On the contrary, tutors can also rate and give comments to users' activities.
- **Contacting, bartering, and communicating at tutor time.** Communication channels (e.g. email and video or audio chat tools) are required for contact information and bartering process between tutors and Go-Lab users. Above all, this communication channel enables tutors to assist Go-Lab users. A video chatting channel is provided to create a real-time, face-to-face like help from tutors to Go-Lab users. Tutors also require a resource upload tool to share learning resources with the users who need tutoring, such as video resources. Screen sharing and documents sharing can further facilitate the help session.
- ***Broadcasting and Recording the tutoring sessions.*** This is a new requirement based on the first-hand use experiences within the Go-Lab community cluster with teachers. A broadcasting session makes it possible to support as many teachers as possible. Recording enables the teachers to repeat the learning content afterwards.
- **Booking tutor time.** Besides contacting tutors via email, a booking functionality is also provided that gives users a clear overview of tutors' availability via a calendar. This booking process can be also cancelled accordingly, if some changes happen later.
- **Recommending tutors.** Go-Lab users will be provided with recommendations of potential experts. This will be combined with social sharing. This requires community support by the Go-Lab ILS experts.
- **Searching tutors.** Go-lab users can search the tutors for certain labs or ILS'. This requirement is also refined to manage tutors' offering sessions. It is useful for users to search for tutors who give help help session in the future.
- **Listing tutors.** A list of experienced tutors is provided per lab and ILS in the Go-Lab Repository.
- Assigning credits to users (nice-to-have). As a teachers' community building measure, users get a certain number of credits when they start using the tutoring platform in order to book tutors' help session. This functionality extends the complete bartering process. Credits could be social media badges, vouchers, and currency.
- Exchange credits among Go-Lab users and tutors (nice-to-have). As a sequence of *assigning credits to users* tutors offer their help sessions against users' vouchers. Also tutors are able to use them to get help from other

tutors if necessary. It is required to be adapted to teachers' community building.

- ***Notify Go-Lab users and tutors.*** Notifications are sent out of the tutoring platform in several cases. Tutors are notified if their sessions are booked or booking cancelled by users. Tutors will be able to send a group emails to all users who have booked the session to keep them posted with information. Users will get a reminder email shortly before the session. Similar to the booking system, the time zones will be taken into account for teachers' convenience.
- ***A forum for tutors and users.*** This is a new requirement aiming at community building. A forum should be established to enable users and tutors to communicate and exchange expertise among themselves, besides email contacting and video chatting.

3.3.3 Components of the Go-Lab Tutoring Platform

The employment of Google services is further specified in this section. And further support to teachers community building is discussed.

3.3.3.1 Google services integration

The connection between the tutoring platform and Google services is improved based on reviews of D4.4. Below is the previous review about the Google service from EC project reviewers:

> D4.4 - There is a reservation on using Google services which forces users to get a Google account, could by the end of the project an open access solution be found without such a constraint.

Considering this review, the tutoring platform should not force users to have a Gmail account. Any mail account connected to Google Plus account is needed to conduct the tutoring sessions supported by Google Hangouts. Any authenticated user is able to book a tutoring session and attend Webinars. So far Go-Lab does not implement another video chat tool from scratch, the Google Hangouts service is in use. The video chatting still requires the Google Plus account.

Integration for the communication between tutors and users employs Google Hangouts for the implementation of the component "Contact & communication" and the component "Single Sign-on".

Since Google Hangouts support 10 users who participate in a session with their web camera and microphone devices, it doesn't support a large-scale broadcasting, such as a webinar. Thus, a further integration of Google Hangouts on Air (HOA) is required to meet with the requests for teachers to attend a tutoring session with a large community of teachers at the same time. The Webinar links are accessible for all users.

Accordingly, tutors can select the help session type whether the help session supports a few teachers to have a close help session or it gives a broadcasting

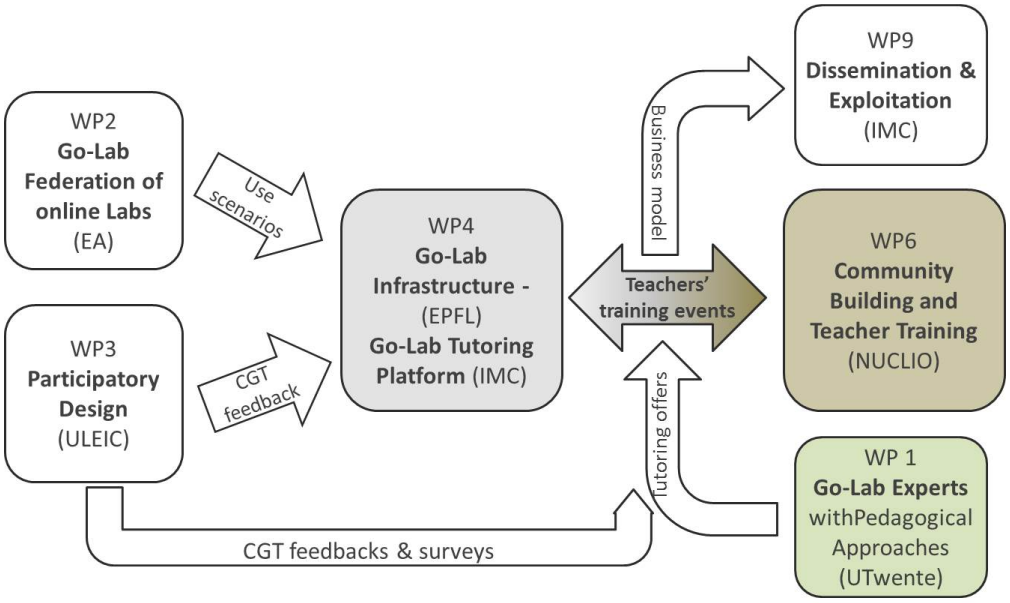


Figure 24. A road map of Go-Lab community building with the Go-Lab Tutoring Platform.

session to a large number of teachers.

Google user login is enabled in the Go-Lab Tutoring Platform for users who don't want to create a new user account on each online platform. The connection to Google API could be configurable by the system administrator in the portal users interface.

3.3.4 Support teachers virtual community building

Figure 24 depicts the role of the tutoring platform to that purpose. It serves as a road map for community building with the tutoring platform. According to the review of Year 2 as mentioned before, support teachers community building should be a main goal of the tutoring platform and more connections from the tutoring platform to other other work packages. The tutoring platform offers the technical infrastructure for teachers, while pedagogical approaches and community events are offered by other relevant work packages such as Work Package 1 and Work Package 6.

Based on this road map, cooperation work with other work packages has been making progresses. We have introduced the tutoring platform to the core group teachers (CGT) at the participatory design workshops by Work Package 3. Cooperation is to be strengthened with Work Package 2 to develop detailed use scenarios with school teachers. Work Package 1 offers pedagogical approaches to conduct the tutoring session with the focus on content and experts.

In use of the tutoring platform, since February 2015, some teachers have attended the tutoring sessions given by WP 2 partner Eleftheria Tsourlidaki. Figure 25 and Figure 26 show the teachers' use of the tutoring platform. One is the tutoring session with a requirement of Google user account. The other is a

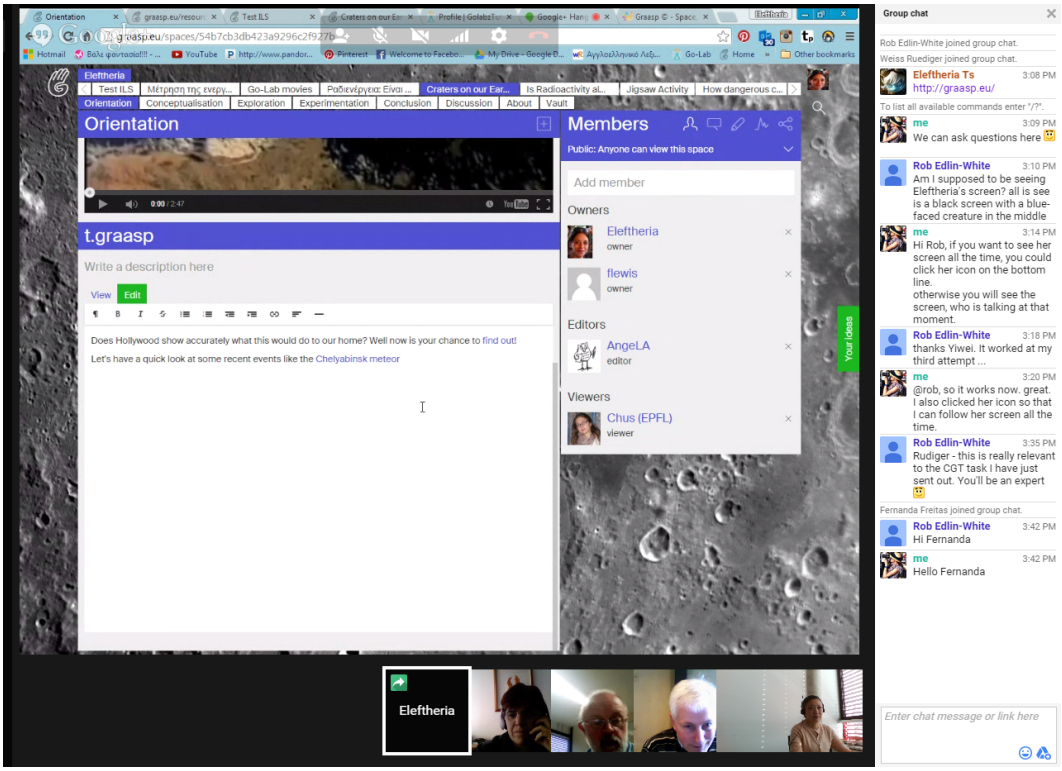


Figure 25. A tutoring session with video chatting to teachers

Webinar in which any user could submit questions and watch the broadcasting without Google user account.

The research results together with WP3 and WP2 will be implemented and applied in teacher community building and teacher training within Work Package 6. This teacher training realisation in the real world is the basic idea of the business model development in Work Package 9.

Work Package of Community Building (WP6) has also proposed the new feature that a forum will be established will contribute to support active communication between the tutor and user community. This features has been specified that the forum will consist discussion topics. Each discussion topic consists of discussion threads given by community users.

Moreover, the tutoring platform should be well linked to the Go-Lab Portal. Thus, the Go-Lab Portal users could find and join the teachers' community easily. More important, from the aspect of a business model for sustainability of the Go-Lab Tutoring Platform, teachers are trained and promoted to become tutors in certain knowledge domains themselves. Go-Lab experts are indispensable to impart the knowledge and offer professional trainings for STEM teachers. These sessions also enable teachers to become experts to offer tutoring sessions to other teachers, which is one of the main purposes of Go-Lab community building. The tutoring platform help teachers break off on this professional development path for teachers' community.

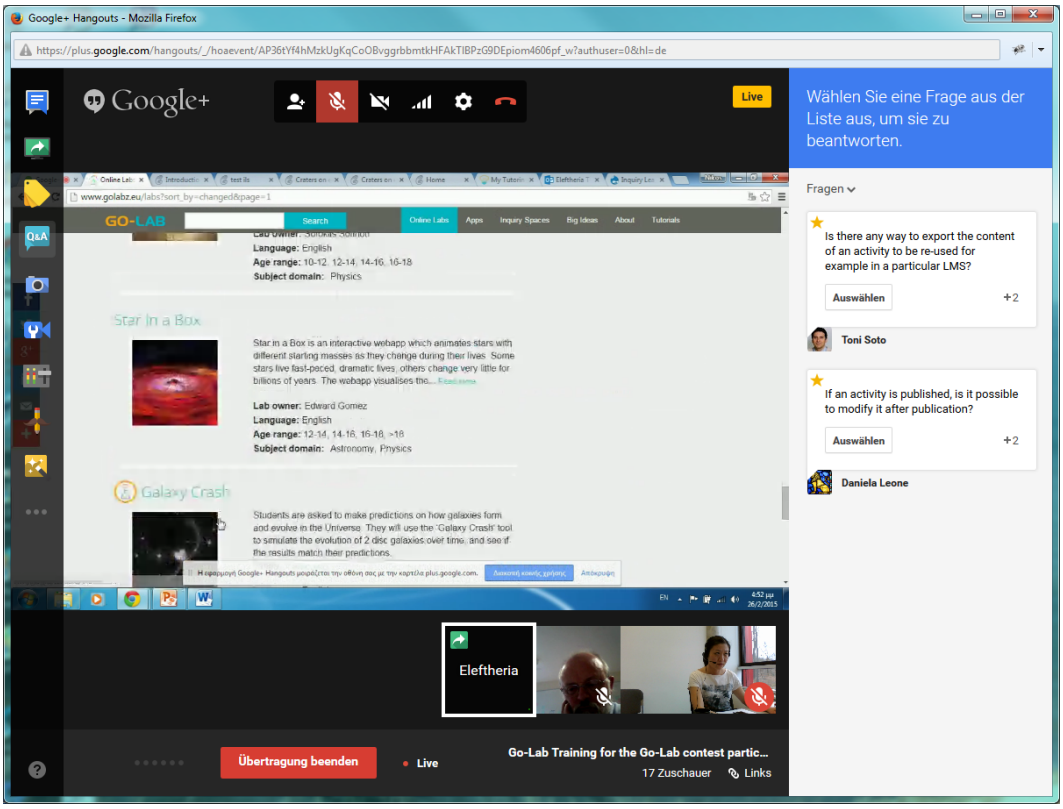


Figure 26. A broadcasting tutoring session to seventeen teachers

4 Conclusion

In this deliverable we have finalized the specifications for the learning analytics, scaffolding, and add-on services in Go-Lab.

The specifications of the add-on services are divided into two parts. The requirements in the initial specifications have been reviewed thoroughly and adapted accordingly. Reviews from the previous deliverables D4.2 and D4.4 are considered for the improvement of the system and platform features. The specification of the Go-Lab Booking System focuses on how to integrate the booking processes distributed in the Go-Lab Portal and Smart Gateway as well as Smart Devices. The specification of the Go-Lab Tutoring Platform has included new features and approaches to support teachers' community building. Evaluations of initial learning analytics apps from the Go-Lab summer schools haven been taken into account to accomplish the final specification for the learning analytics and scaffolding services. As an outcome, different approaches to support the different stakeholders have been followed, particularly the specification of a teacher dashboard as a composition of LA apps for teachers, but also apps to support learners, e.g., through presenting higher representations of learning processes.

Additionally, to foster the sustainability of the learning analytics infrastructure, emphasis has been put on shaping development interfaces and APIs. A rich documentation is available to outline all the important specifications and APIs to develop new learning analytics apps. In this deliverable, several examples of how to develop different kind of apps and services have been presented.

As Part 1, to brief the reviews received at the end of Project Year 2 (cf. the first paragraph in Section 3.3), we list our approaches and measures on the tutoring platform accordingly.

1. Review: Worries on the Go-Lab sustainability and a close connection to pedagogical work package were addressed. Measures: we have elaborated a road map to gain expertise from collaboration with various other Go-Lab work packages (cf. Figure 24) beyond WP4.
2. Review: Too late implying adaptation costs should be avoided. Measures: based on the road map we have started to work together with other work packages actively. The request to establish a forum for the tutoring platform is proposed by the community work package (WP 6).
3. Review: Future activities of the tutoring sessions should be precisely documented. Measures: Tutoring sessions have run on the tutoring platform, which is mentioned in the latest deliverable D7.5. It will be further reported in D4.8 and in other deliverables especially in Work Package 6.
4. Review: The initial recruitment of experts is important due to the lack of teachers' skills. Measures: Again recalled the road map, pedagogical experts from Work Package 1 and Work Package 2 have been giving tutoring sessions to teachers at this bootstrap phase. Feedback from teachers will

be reported in D4.8 and other deliverables.

As Part 2, the specification of the booking system gives a detailed guide for the final release. So far, many discussions and expertise exchanges have been conducted between Go-Lab technical experts and lab owners, which leads to completion of the specification. At the same time, we also keep in mind that cooperation and communication with lab owners are the key issues. Close connections to lab owners will be further maintained to create lab owners' lab calendars and enable teachers to book those remote labs.

This deliverable offers the final specifications for the final system releases which will be delivered in D4.8 (M36).

References

Breuer, R., Klamma, R., Cao, Y., & Vuorikari, R. (2009). Social network analysis of 45,000 schools: A case study of technology enhanced learning in europe. In U. Cress, V. Dimitrova, & M. Specht (Eds.), *Learning in the synergy of multiple disciplines* (Vol. 5794, p. 166-180). Springer Berlin Heidelberg.

Cao, Y., Govaerts, S., Dikke, D., Faltin, N., & Gillet, D. (2014). Helping each other teach: design and realisation of a social tutoring platform. In *Journal of immersive education (jied) - proceedings of 4th european immersive education summit, immersive education initiative, austria.*

Cherradi, F. (2015). *Web-based bartering platform* . Trier University of Applied Sciences, Umwelt Campus Birkenfeld.

Clariana, R., Engelmann, T., & Yu, W. (2013). Using centrality of concept maps as a measure of problem space states in computer-supported collaborative problem solving. *Educational Technology Research and Development*, *61*(3), 423-442. Retrieved from `http://dx.doi.org/10.1007/s11423-013-9293-6`

de Jong, T., van Joolingen, W. R., Giemza, A., Girault, I., Hoppe, U., Kindermann, J., ... van der Zanden, M. (2010). Learning by creating and exchanging objects: the SCY experience. *British Journal of Educational Technology*, *41*, 909–921.

De Wever, B., Schellens, T., Valcke, M., & Van Keer, H. (2006, January). Content analysis schemes to analyze transcripts of online asynchronous discussion groups: A review. *Comput. Educ.*, *46*(1), 6–28.

Duval, E. (2011). Attention please! learning analytics for visualization and recommendation. In *1st international conference on learning analytics and knowledge (lak 2011)* (pp. 9–17). Banff, Canada: ACM.

Facca, F. M., & Lanzi, P. L. (2005). Mining interesting knowledge from weblogs: a survey. *Data Knowl. Eng.*, *53*(3), 225-241.

Ferguson, R. (2012). Learning analytics: drivers, developments and challenges. *International Journal of Technology Enhanced Learning*, *4*(5), 304–317.

Göhnert, T., Ziebarth, S., Malzahn, N., & Hoppe, H. (2014). Enriching (learning) community platforms with learning analytics components. In N. Baloian, F. Burstein, H. Ogata, F. Santoro, & G. Zurita (Eds.), *Collaboration and technology* (Vol. 8658, p. 177-184). Springer International Publishing.

Harrer, A., & Göhnert, T. (2015). Integrated representations and small data: Towards contextualized and embedded analytics tools for learners. In *Proceedings of the fifth international conference on learning analytics and knowledge* (pp. 406–407). New York, NY, USA: ACM. doi: 10.1145/2723576.2723665

Harrer, A., Malzahn, N., Zeini, S., & Hoppe, H. U. (2007). Combining social network analysis with semantic relations to support the evolution of a scientific community. In C. Chinn, G. Erkens, & S. Puntambekar (Eds.), *Mice, minds, and society - the computer supported collaborative learning (cscl) conference 2007* (p. 267-276). International Society of the Learning Sciences.

Hoppe, H. U., Engler, J., & Weinbrenner, S. (2012, 07). The impact of structural characteristics of concept maps on automatic quality measurement. Sydney, Australia.

Jimoyiannis, A. (2010, November). Designing and implementing an integrated technological pedagogical science knowledge framework for science teachers professional development. *Computers & Education*, *55*(3), 1259–1269.

Laat, M. D., Lally, V., Lipponen, L., & Simons, R.-J. (2007). Investigating patterns of interaction in networked learning and computer-supported collaborative learning: A role for social network analysis. *I. J. Computer-Supported Collaborative Learning*, *2*(1), 87-103.

Liddo, A. D., Shum, S. B., Quinto, I., Bachler, M., & Cannavacciuolo, L. (2011). Discourse-centric learning analytics. In *Lak* (p. 23-33).

Manske, S., Hecking, T., Bollen, L., Gohnert, T., Ramos, A., & Hoppe, H. (2014, July). A flexible framework for the authoring of reusable and portable learning analytics gadgets. In *Advanced learning technologies (icalt), 2014 ieee 14th international conference on* (p. 254-258). doi: 10.1109/ICALT.2014.80

Siemens, G. (2012). Learning analytics: envisioning a research discipline and a domain of practice. In *Proceedings of the 2nd international conference on learning analytics and knowledge* (pp. 4–8). New York, NY, USA: ACM.

Southavilay, V., Yacef, K., Reimann, P., & Calvo, R. A. (2013). Analysis of collaborative writing processes using revision maps and probabilistic topic models. In *Lak* (p. 38-47).

Suthers, D. D., Lund, K., Rosé, C. P., Teplovs, C., & Law, N. (2013). Productive multivocality in the analysis of group interactions. *New York, NY: Springer. CSCL book series. doi*, *10*(1007), 978–1.

Zaıane, O. R., & Luo, J. (2001). Web usage mining for a better web-based learning environment. In *Proceedings of conference on advanced technology for education* (pp. 60–64).

Zhang, J., Ackerman, M. S., & Adamic, L. (2007). Expertise networks in online communities: structure and algorithms. In *Proceedings of the 16th international conference on world wide web* (pp. 221–230). New York, NY, USA: ACM.