Mobile Inquiry Learning in Sweden: Development Insights on Interoperability, Extensibility and Sustainability of the LETS GO Software System

Bahtijar Vogel*, Arianit Kurti, Marcelo Milrad, Emil Johansson and Maximilian Müller
Department of Media Technology, Linnaeus University // bahtijar.vogel@lnu.se // arianit.kurti@lnu.se // marcelo.milrad@lnu.se // emil.johansson@lnu.se // maximilian.muller@lnu.se

*Corresponding author

ABSTRACT
This paper presents the overall lifecycle and evolution of a software system we have developed in relation to the Learning Ecology through Science with Global Outcomes (LETS GO) research project. One of the aims of the project is to support “open inquiry learning” using mobile science collaboratories that provide open software tools and resources, and participation frameworks for learner project collaboration, mobile data and media capture, publishing, analysis, and reflection. The primary focus of this paper is to report on our technical development, insights and knowledge gained during the past four years. Technical implementations and the prototypes developed in this project have been tested across several educational trials conducted in Sweden and abroad with more than 400 learners. Insights and knowledge gained from these activities verify that learners’ requirements were adequately addressed while satisfying their needs. The outcomes and results of our efforts provided us with a better understanding with regard to which software engineering processes and approaches can be used to address and support the complex requirements that emerge in novel mobile learning scenarios. Thus, the results discussed in this paper provide deeper insights into the importance of properly addressing issues related to interoperability and extensibility in order to develop software solutions to support mobile learning that are sustainable and endurable over time.

Keywords
Mobile learning, Inquiry-based learning, Software lifecycle, User-centered development, Interoperability, Extensibility, Sustainability

Introduction

Web technologies are enabling Internet applications and services to become easily integrated in interactive systems (Holmberg, Wuenhsche, & Tempero, 2006). Thus, the web is gradually becoming a “central computer” that helps to connect diverse computing and data resources and people (Liang, Croitoru, & Tao, 2005; Giusto, Iera, Morabito, & Atzori, 2010). The evolution of these web developments combined with sensor and interactive technologies provide new possibilities for the implementation and deployment of software applications to support a wide variety of human activities.

Mobile and web technologies and applications provide new possibilities for augmenting learning activities. These are Technology-Enhanced Learning (TEL) activities that can be spatially distributed and can incorporate different physical and environmental sensor data (Wu, Yang, Hwang, & Chu, 2008). There are different mobile, web and sensor-based technologies that provide new perspectives on how learning activities can be embedded in different settings and across contexts (Chang, Wang, & Lin, 2009). One innovative aspect of these new learning landscapes is the combination of learning activities to be conducted across different educational contexts such as schools, nature and science centres/museums, parks, and field trips (Kukulska-Hulme, Sharples, Milrad, Arnedillo-Sanchez, & Vavoula, 2009). In these technology rich and dynamic learning environments learners make use of a wide range of devices and applications and the notions of system interoperability and extensibility, become central in order to successfully fulfill the requirements posed by the different educational activities and the learners. Especially, since these aspects directly influence learners’ satisfaction and experiences with regard to the applications and systems they use and also directly affect how tools and applications are adopted, appropriated and sustained over time.

In our Learning Ecology with Technologies from Science for Global Outcomes (LETS GO) collaborative international project (2008-2012), we have been developing, implementing, studying and scaling up novel ways for fostering secondary school student learning in teams for ecological and environmental sciences (Spikol, Milrad, Maldonado, & Pea, 2009). During the last 4 years we have been working with the design, development and implementation of web and mobile services that integrate geo-sensing, multimedia communication and interactive visualization techniques in specific ecology learning scenarios. Our goal has been to create mobile science inquiry
collaboratories (Pea, Milrad, Maldonado, Vogel, Kurti, & Spikol, 2012) with teachers, learners and developers and domain scientist on topics related to water and soil quality, ecosystems and biodiversity.

One of the main objectives of the LETS GO project was to develop a robust software system including a wide range of applications and services to support educational activities that promote collaborative scientific inquiry as students formulate questions and hypotheses, and collect, analyze, discuss and compare data while studying problem topics in environmental sciences. All our software solutions have been conceived having in mind how to support all these processes. In this paper we present and discuss the overall lifecycle and evolution of the software system we have developed during the last four years. Our choice to focus on these specific aspects is guided by the challenge of how to address those problems related to the scalability and interoperability of mobile learning applications. Thus, the main question we are trying to answer in this paper can be formulated as following: How can software engineering processes support the functional requirements posed by current mobile learning scenarios and applications? The insights and knowledge gained during our research efforts are closely related to the issues of interoperability, flexibility and extensibility of the LETS GO software system and were identified during iterative user-centred development cycles. Developing sustainable mobile applications that can cope with the changing demands of dynamic learning environments requires new knowledge and approaches. The results presented and discussed in this paper provide some new perspectives in this direction.

The remaining of the paper follows with a presentation of the motivation behind our research efforts, as well as the initial requirements, to continue after with an overview of the LETS GO project and its related activities. The following section presents the details of our design, technical solution and the evolutionary stages that were carried out as a part of this development to continue with the Lessons Learned section where we reflect upon the activities we have conducted during these four years of development. At the end, we provide our main conclusions and discuss possible lines for future research.

**Motivation and initial requirements**

The initial requirements elicitation with stakeholders emerged from a workshop with teachers involved in the LETS GO project that took place in the fall 2008. Different activities in this workshop helped to identify the need to integrate geo-location and environmental sensing, visualization, and Web 2.0 mashup technologies, as part of a broader educational scenario. These requirements identified the need to support “open inquiry learning” for having access to diverse sensor data, live mapping tools, interactive data visualization and collaboration tools, and additional learning resources. Another requirements related to usability, include low cost, using open standards, multiple application support, and support for different types of collaboration modes and contexts (Spikol, Milrad, Maldonado, & Pea, 2009).

Trying to match these initial requirements brought up a number of challenges that concern software tools for supporting inquiry-learning activities. A survey of the literature and existing approaches to support inquiry science learning conducted at the beginning of the project indicated that there were no existing software solutions that could cope with all these requirements at the same time (Vogel, Spikol, Kurti, & Milrad, 2010). More recently, Sun & Looi (2013) report on a review of different web-based science learning environments for collaborative inquiry. The analysis of the results indicate that even those systems discussed in their paper do not cope with the kind of requirements we are addressing in our work. Already at the early stage of our project, those aspects related to the issues of interoperability and extensibility of the system to be developed were identified as one of the central challenges in terms of “building new technologies or further developing existing technologies to create novel possibilities for supporting human activities” (Tchounikine, 2011). Some of the processes that learners need to be actively involved during inquiry learning activities are to problematize, demand, discover and refine, and apply new knowledge and skills to solve complex problems (Edelson, Gordin, & Pea, 1999). Therefore, our primary focus in this project was to facilitate the integration of proper tools (both hardware and software) and services for supporting inquiry based learning activities.

According to Knapp and Barrie (2001), field trips are important to effectively learn about environmental science and they should be actively promoted. It is suggested that field trips can be helpful to generate relevancy to classroom learning when connected with the outdoor environment. For students, such an approach may raise the interest in and aspirations for science-related careers (Rudmann, 1994). The data collected in such field trips play an important role.
for analysis, and hence should be saved and carried back to the classroom. Presenting and analyzing these data using visualization tools may help to increase learners’ understanding of complex subject matter.

Reflecting upon our current knowledge and experiences from the field of TEL, two important issues can be identified for supporting environmental inquiry science learning including outdoors and in classroom activities:
1. Providing technological support (in terms of portable instruments and sensors for data collection and software) for field trips activities that include collecting data, and
2. Providing technological support for classroom activities that include visualizing, exploring, analyzing, discussing and reflecting upon the data collected in the field.

Hence, the system support for these kinds of activities needed to include functionalities for mobile data collection and web-based tools and applications for interactive visualizations. These aspects were also in line with the theoretical aspects of scientific inquiry thinking that suggest that this kind of system support has the potential to increase learners’ engagement and curiosity (Pea, 2002). Thus, based on these different requirements we have developed a variety of software tools and solutions to address these different challenges. A detailed description of this work can be found at Vogel et al. (2010), Vogel et al. (2011) and Vogel (2012).

Figure 1 below provides an initial overview of the educational settings related to the LETS GO project and presents some of the initial requirements. Moreover, this figure maps the key processes of inquiry learning activities related to data collection and interpretation, exploration and reflection, drawing conclusions and communicating the results (Edelson, Gordin, & Pea, 1999; Linn, & Eylon, 2011).

As illustrated in this sketch, the initial requirements that guided our research efforts can be specified in terms of the following:
- Mobility of the users/learners
- Distributed environments,
- Service-oriented systems,
- The need for reflection on the collected data and activities, and
- Interactive collaborative technologies.

Some of these requirements are also in line with the current key technological trends identified in the field of TEL: mobile and cloud computing, visual data analysis, web technologies and geocoded data, smart objects and open content, as pointed out recently in the literature (Johnson, Levine, & Smith, 2009; Johnson, Levine, Smith, & Stone, 2010; Johnson, Smith, Willis, Levine, & Haywood, 2011). Indeed, the latest Horizon Report (Johnson, Smith, Willis, Levine, & Haywood, 2011) points out that many technologies used in the field of TEL are increasingly becoming
cloud-based and decentralized. From a pure technological perspective, Hoppe (2009) claims that one of the main challenges we are facing involves the need for integration of diverse technological resources in broader educational scenarios. Therefore, these trends once more reaffirm that the issues related to interoperability and extensibility become central for the integration of diverse technological resources for supporting educational activities. In the coming section we provide an overview of the LETS GO activity flow in order to better understand the interplay between the different learning activities and the technological support.

LETS GO activities and testing

During the four years of the project more than 400 students have been involved in different type of learning activities. These activities included classroom lessons, field trips and lab work and included data collection in the field, taking images and notes, as well as data visualization and discussions in the classroom (see figure 2).

Usually, the participants in the different activities were either students from K-12 schools in Växjö, Sweden or undergraduate students (the teacher training program at Linnaeus University). As part of the environmental science curriculum, they investigated topics related to soil quality (woodland ecology) and water quality in the surrounding lakes. None of the participants in all these activities had prior knowledge regarding how to use the technologies we developed. In our latest pilot activity that took place at the Potomac River in the USA (September 2012), teachers from both Sweden and USA were involved.

Figure 2. Different learning activities and the technologies in use

Figure 3 below gives an overview of the learning activity flow and how the different phases of the students’ inquiry process were supported. It should be noticed that these activities where designed according to the different stages of inquiry based-learning as suggested by Edelson, Gordin, & Pea (1999) & Linn & Eylon (2011). Furthermore, these different learning activities have been integrated with their regular curricula in Environmental Science courses at the different schools. A typical LETS GO learning activity usually included workshops for the students to get familiarized with specific subject matter and central concepts and ideas associated with the inquiry learning process. These activities usually comprised six to eight lessons over a period of five weeks starting with the introduction of inquiry process where basic concepts of the activity were introduced; students discussed the initial questions given to them about a specific topic (e.g., water quality). This activity was followed by the preparation for investigation and experiments to be conducted using different technologies (proves, data loggers, mobile applications for data collection in the classroom). Additionally, users conduct field experiments at a local environment and collect samples for lab analysis (see Figure 2a). The data collected using the mobile data collection tool were geo-tagged content and sensor data (usually pH, dissolved oxygen, temperature, conductivity, moisture, etc. depending on the type of the activity). The learning activity usually ended with a discussion about their findings from the field and lab work and an overall class discussion and reflection by using the web visualization tool (see Figure 2b), which tailored different geo-tagged sensor data and digital content collected using mobile data collection tool. In average, (depending of their course schedule) the entire activity was conducted over the period of four weeks across five lessons units. The logistics, as well as the time period of these activities are illustrated in figure 3 below. As it is presented, each one of these lessons units generated a set of functional requirements that the system should support, namely: sensor support, mobile data collection and interactive data visualization. Furthermore, since these activities (mainly for K-12 students) were part of the regular curricula, students were asked to submit short reports on the outcomes of their efforts and reflections after each unit, concluding with a final test at the end of the activity.
Throughout these activities, we videotaped different sessions for later analysis and some of the researchers from our group used a systematic observation sheet during field and lab sessions.

During the lifetime of the project, we actively tested all our developments with school students where we combined classroom and field trips activities. The user trials (prototype testing) allowed testing the software application throughout five development iterations on authentic settings and dynamically changing environments, while new requirements continuously emerged in these activities. These iterations include the release of a prototype and its active testing with the users/learners. The initial two prototypes have been of throwaway type with a single iteration stage each. The last prototype has been of evolutionary nature that evolved through three following iterations cycles. Details regarding these prototypes as well as development iterations are presented in the Table 1.

Table 1 below, provides a detailed overview of the LETS GO field activities conducted since May 2009 until our latest activity conducted in September 2012. This table provides an overview about learner generated content, and the records that were stored in our repositories. It provides a summarized view on how many samples/records learners collected during these activities, as well as number of pictures they stored in our database and server resources. Furthermore, it also provides a rough overview about the number of users that used our software system so far, including information about location of the trials, school they belonged, and type of the activity they were engaged.

<table>
<thead>
<tr>
<th>Protype</th>
<th>Development</th>
<th>Deployment and Testing</th>
<th>Organisation</th>
<th>Users</th>
<th>Location</th>
<th>Records in Database</th>
<th>Images</th>
<th>Activity</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Prototype</td>
<td>2009-May</td>
<td>Katedralskol</td>
<td>14</td>
<td>near Katedralskol, Varpa, Sweden</td>
<td>24</td>
<td>48</td>
<td>outdoor/indoor</td>
<td>soil quality</td>
<td></td>
</tr>
<tr>
<td>2nd Prototype</td>
<td>2010-June</td>
<td>Teleborg Skolan</td>
<td>25</td>
<td>Teleborg Skolan, Varpa, Sweden</td>
<td>97</td>
<td>182</td>
<td>indoor</td>
<td>water quality</td>
<td></td>
</tr>
<tr>
<td>3rd Prototype</td>
<td>2010-April</td>
<td>Kronberg Skolan</td>
<td>60</td>
<td>Kronberg Skolan, Varpa</td>
<td>139</td>
<td>318</td>
<td>indoor</td>
<td>water quality</td>
<td></td>
</tr>
<tr>
<td>4th Prototype</td>
<td>2011-October</td>
<td>Katedralskol</td>
<td>40</td>
<td>Varpa, near Katedralskol</td>
<td>31</td>
<td>55</td>
<td>outdoor</td>
<td>soil quality</td>
<td></td>
</tr>
<tr>
<td>5th Prototype</td>
<td>2012-April/May</td>
<td>Kronberg Skolan</td>
<td>75</td>
<td>Varpa, Varpa, Sweden</td>
<td>130</td>
<td>267</td>
<td>outdoor</td>
<td>water quality</td>
<td></td>
</tr>
<tr>
<td>6th Prototype</td>
<td>2012-May</td>
<td>Lars Jagger</td>
<td>25</td>
<td>Baltic Sea, Kalmar, Sweden</td>
<td>8</td>
<td>32</td>
<td>outdoor</td>
<td>water quality</td>
<td></td>
</tr>
<tr>
<td>7th Prototype</td>
<td>2012-September</td>
<td>National Geographic Society</td>
<td>18</td>
<td>Potomac River, Washington DC, USA</td>
<td>18</td>
<td>44</td>
<td>outdoor</td>
<td>water quality</td>
<td></td>
</tr>
</tbody>
</table>

Table 1. LETS GO Activities and related data
The evolution of our software system

The LETS GO software system has gone through an evolutionary prototyping approach through the past four years in order to become a stable and robust platform for mobile data collection, aggregation and data visualization. During the prototyping efforts, we extensively used two different development principles: evolutionary prototyping and throwaway prototyping (Sharp, Rogers, & Preece, 2007). The iterative user-centred development cycles of the software system are mapped to our developments, following these two principles of prototyping:

**Throwaway prototyping**—considers creating the basis of a final product, which is eventually thrown away; however, it remains valuable to construct further evolving ideas related to the final product (1st and 2nd prototype).

**Evolutionary prototyping**—considers the evolution of a prototype toward a robust final product (3rd prototype). We applied both these principles throughout our development process.

These two principles of prototyping led to the development of an application that was later used in testing. The design and implementation of the three prototypes was made possible by an evolutionary process through several iterations (as already introduced in Table 1 above). Two initial prototypes have been throwaway type while the last one has been an evolutionary prototype.

Figure 4 presents the timeline overview of the different iterative development cycles and stages of the three software prototypes. During these development efforts, we heavily utilized service-oriented approaches for supporting TEL activities in the context of inquiry learning. These prototypes of a software system evolved from being proprietary applications towards combining several Internet-based services to process and visualize the geo-temporal data collected using mobile data collection tools and web. In the first development stage, we mainly dealt with the integration challenges of different technological resources. In addition, the first software prototype has been mainly implemented using static forms for mobile and desktop enabled visualization features and did not provide real time data representation (and thus was throwaway prototype). During the second development stage, we continued with our integration challenges, however, due to the evolution of requirements, the challenge of interoperability across diverse technological resources arose. Thus, the second prototype resulted in a combination between more dynamic forms for mobile (XForms), desktop and web technologies and included initial cloud services (Vogel, 2011). This combination enabled data collection in both, online and offline modes and real time data representation. Finally, during the third development stage we continued to address the interoperability issues, thus the last prototype of the visualization completely relied on an Internet based environment (real time and online). The final effort during this process was to conduct a user testing study (Vogel, Kurti, Milrad, & Kerren, 2011). The three development stages made our software system more robust as a product compared with the earlier implementation. A usability study was conducted for assessing the web-based visualization tool since it aggregated and presented the entire data collected using mobile devices. Assessment and testing of the web-based visualization tool was an important issue, in order to identify usability aspects that resulted in a number of concrete suggestions for the further enhancement and improvement of. These suggestions were later used and translated into requirements for further development.
The system architecture and implementation

During the efforts mentioned in the previous sections, the technical developments have evolved and changed rapidly, such as in terms of the design aspects, technology choices and implementation, as well as software & hardware components. Despite the rapid changes of such technologies these developments have not been reflected in the changes of the architecture of our software system (Vogel, Kurti, Milrad, & Mikkonen, 2012). The components identified in the initial architecture have been proven resilient to these dynamic changes and requirements. The results described in this paper are an effort to try to tackle some of the challenges described earlier and are an evolution of the efforts we conducted in one of our previous work (see Vogel, Spikol, Kurti & Milrad, 2010). During our earlier explorations and developments, we have proposed and implemented a system architecture that consists of five different blocks aiming to provide logical divisions between the different resources of our system. Figure 5 below illustrates the component view of the system architecture and its potential for expandability with other technologies and external systems.

In the architecture presented in Figure 5, resources are organized into different building blocks that integrate sensors, mobile data collection units, from the server side the data aggregation components used as data and content storage, and visual representation components that utilize diverse web APIs using external services block of this architecture (Vogel, 2012).

![Figure 5. System architecture including main resources and components](image)

The design and implementation of our mobile data collection tool followed with the integration of an open source Java based project into our software system (Anokwa, Hartung, Brunette, Borriello, & Lerer, 2009). This project/solution supported the use of a particular open standard called XForm. XForm is a standard based on a W3C recommendation that is used to build web forms for the easy exchange of data across platforms and devices using XML as the data format. In our case, the design of the forms for data collection used in our mobile applications was developed following the requirements identified from stakeholders, as introduced above. The solution we decided to adopt is based on the Open Data Kit (ODK) which supports various types of data and content inputs, including text, audio, pictures, video, visual codes and GPS and makes it possible to annotate the collected sensor data and content with location metadata (Anokwa, Hartung, Brunette, Borriello, & Lerer, 2009). The use of XForms facilitated data interoperability across a diverse range of devices and applications that compose our system. We have therefore developed several mobile forms that rely on the use of open standards, which provided us with flexibility, fast development and easy adaptation and integration of technological resources for different scenarios (user trials). Figure 6 shows the screen shot of LETS GO data collection tool.

In terms of software development, our major efforts were allocated for the implementation of the web-based visualization tool. The latest version of this tool enables the visualization of different types of geo-tagged content and sensor data collected using the mobile application described above. The web visualization tool utilizes APIs that provide multiple visual representations of the data set available in our repository. These representations allow users to actively interact with graphs, maps, images, and data tables. An initial version of this tool was implemented...
completely in AJAX. Spite a user friendly interface and positive feedback from the users, we experienced some performance problems related to loading and processing huge amounts of data using AJAX (Vogel, 2011).

These latest drawbacks inspired the latest version of our visualization tool that has been entirely developed using Google Web Toolkit (GWT). Our latest version of web based visualization tool called GreenLab (see Figure 7) addressed a lot of the requirements generated during our testing efforts by using iterative cycles with users. This tool has become more stable that the previous AJAX version and it has a lot of features for mapping the data automatically, filtering the data based on different criteria, etc. The data visualization process in Green Lab is divided into two stages; first Green Lab selects the type of data collected during the activity with the mobile devices; and second, it presents and visualizes the collected data. In the first stage, Green Lab loads each data type and presents them as clickable buttons. Each type can contain several forms (XForms), which will be loaded after the user has selected a specific activity. In the second stage, Green Lab presents the data retrieved from each form. The filtering of data in Green Lab is located in a panel and the filtering mechanism triggers all active visualization to update, presenting only a certain part of the data set. Each checkbox as such listens for user clicks, triggering an event to update the views. The filtering panel view contains three kinds of options. The first filtering view is based on Organizations (namely participating schools) that follows with the Groups that belong to that organization. Selecting an organization will first filter the data on selected option and also present all groups for that organization, which also allows filtering specific groups. The Attributes filtering view is constituted from each attributes that can be filtered accordingly, by also selecting multiple attributes. Moreover, a single attribute can be filtered also by their values. The Dates filtering view can be used to choose From-To dates to set the time period the user is interested to investigate.

Figure 6. LetsGo mobile data collection based on ODK

Figure 7. Green Lab – Web visualization tool

Green Lab contains three different visualization views for presenting the data, which are all resizable. The main visualization view starts up by presenting the data as a table, but has the possibility to switch to bar-, columns-, line-,
or an area-chart by navigating to the icons in the header. The second view contains a Map view, which locates and visualizes all the geo-tagged data collected by the mobile application. The last view contains only a Scatter Plot view, which allows filtering of two kinds of data attributes, by comparing them.

The design and implementation stages carried out during the past four years facilitated the identification of the main features in terms of sustainability of the LETS GO system. The three main salient features that we have identified during our design and implementation efforts can be enumerated as following: (1) Interoperability of the software and hardware components (2) Extensibility of the visual representation forms, and (3) Sustainability of the software solution. A more detailed view of these lines of action based on what we learned during the last year developing the LETS GO system are presented in the following sections.

Lessons learned

Proprietary software solutions are deployed extensively through multiple platforms such as the web, mobile devices and desktop applications. The use of diverse standards brings new challenges when it comes to flexibility, interoperability, customizability and extensibility of different components that are part of software systems. Continuous evolving web and mobile technologies combined with the changes of the environment result on dynamic and complex requirements that become extremely challenging. As indicated earlier, this paper aims to tackle some of these problems by providing deeper development insights into the issues of interoperability, extensibility and sustainability related to the LETS GO software system and its evolution during the last four years. These insights were gained during iterative user-centred development cycles of a software system designed and implemented for fostering collaborative science learning activities.

Interoperability of the software and hardware solutions

The notion of interoperability constitutes one of the most important principles in system integration (Zeng & Quin, 2008) and refers to "the compatibility of two or more systems such that they can exchange information and data and can use the exchanged information and data without any special manipulation" (Taylor & Joudrey, 2003). During the development efforts of the LETS GO software system, we have in practice been able to tackle two out of four interoperability categories as introduced by Sheth (1998), namely:

- Syntactic interoperability: differences in data formatting.
- System interoperability: heterogeneous systems and applications.

Despite considerable research efforts, achieving interoperability of various sensor readings and mobile devices and various web services has remained an open issue. In connection to syntactic interoperability our focus was related toward making use of open standards for data exchange such as XML, XForms and JSON. While from a system interoperability perspective, the focus was on making use of open software tools with cloud-based services for matching our requirements.

The features of the cloud environment and services made our system more flexible. Initially, traditional desktop-based integrated development environments were employed and the development then gradually moved towards a mashup-pattern that combined different service-oriented approaches. One identified issue was that the rapid speed and evolution of software and web technologies affected the development process and the application itself. From heavily using Internet based services, such as cloud environments and due to the problems encountered from such services, we started to deploy all our developments into our local repositories and environments. By migrating the LETS GO system to our local environment we mitigated the risks and uncertainties of the cloud development environments. Furthermore, this created new opportunities to closely address the interoperability issues of the software components comprising the LETS GO system. During the last year of development we specifically addressed interoperability issues in the “Mobile Data Collection” component and “Data Aggregation” component.

The interoperability issues in the “Mobile Data Collection” component dealt with both hardware and software issues. Since the current mobile data collection tool was tailored only to Android smartphones and having in mind the emergence of HTML5 and CSS3 as well as multiple mobile cross-platform frameworks (such as PhoneGap, Titanium etc.), we have started expanding the mobile collection tool toward iOS and Windows Mobile platforms.
The purpose of this effort will be to widen the base of the eligible devices to be used as mobile data collection tool (so we do not pose any limitations for the use of the mobile application in schools) as well as making them fully interoperable with the rest of the LETS GO system.

Interoperability issues in the “Data Aggregation” component have been addressed from the perspective of data export capabilities. The idea for such development was to make the LETS GO system open and interoperable with similar tools developed in other research projects. The Data Export component we have developed can be described as a middleware that prepares the surveys data stored on the ODK Aggregate server. The main application is a Java servlet that processes calls from clients and replies either a list of available “SurveyTypes,” links/URLs to all the stored forms and their submissions, or the actual data of the submissions. These calls can be processed in the form of a Web API, where the data is responded as JSON. This enables that all the data aggregated in the ODK aggregate becomes available for export using this servlet and in a JSON format. In the current version of the DataExportServlet all the data is read and exported as the clients upload it and stored in the database.

Beside these efforts we have also been working on the development of the form rules as guidelines for design of the XForms for mobile data collection in a form of naming conventions. These naming conventions developed enable mapping of the collected data dynamically to the visualization tool by the form designed based on our guidelines. All these developments enabled our LETS GO system to be fully interoperable with different mobile devices as well as new visualization tools and services.

**Extensibility of the software and system architecture**

The massive use of mobile and web technologies for data collection purposes produces vast amounts of data. This is another challenging task that requires attention while trying to make sense of all the data generated by users. Therefore, as the amount of available data continues to grow, conceptualizing and developing new interactive tools for visualization becomes an important task to tackle these challenges, for, e.g., seeking new ways of presenting and sorting appropriate and relevant data, or managing and analyzing information (Ackerman & Guiz, 2011). Different visual representations can provide different insights to users by enabling them to observe data in context, to analyze these data and to draw different conclusions by using different analytical approaches (Eiselle, & Weiskopf, 2009; Sedig, Liang, & Morey, 2009). The extension of web-based visualization approaches, along with new forms of interactive collaborative technologies, is constantly growing (Sedig, Liang, & Morey, 2009). Lately, TEL researchers have been taking advantage of different interactive visualization techniques and tools (Linn, & Eylon, 2011). Research in this area indicates that visualizations have the potential to improve learning outcomes, especially related to inquiry science learning (Johnson, Levine, Smith, & Stone, 2010; Edelson, Gordin, & Pea, 1999; Linn, & Eylon, 2011; Pea, 2002). Moreover, interactive visualizations support and increase students’ engagement in scientific inquiry (Linn, & Eylon, 2011; Pea, 2002). In the scope of our work, “learning through collaborative visualization” refers to developments of “scientific knowledge that is mediated by scientific visualization tools in a collaborative learning context” (Pea, 2002). The latest version of the LETS GO system allows for integrating new interaction features provide by multitouch enabled devices and gesture based interaction in a way that we can expand the interaction modes in which learners work with the visualizations.

Recent developments of our LETS GO system include the implementations of two prototypes using gesture based interaction supported by the use of the Microsoft Kinect (Vogel, Pettersson, O., Kurti, & Huck, 2012) and touch enabled interactions facilitated by the used of the Samsung SUR-40 tabletop computing system (Müller, 2012). Both these prototypes are fully functional and make use of the data already stored at the LETS GO repository. The initial benefit seems to be the fact that these two new interactions paradigms promote collaboration among users while reflecting upon collected data. Figure 8 below illustrates the Natural User Interface (NUI) for the Green Lab application.

NUI Green Lab is a visually driven explorative interactive visualization tool. The tool focuses on a graspable presentation of the geo-tagged environmental data, collected during outdoor activities using mobile devices, in form of digital maps, charts, and images. The application provides a multi-user interface facilitating the synchronous collocated collaboration of at least two users. The interaction makes use of multi-touch interactions on the SUR-40 tabletop computing system and in-air gestures facilitated by the Microsoft Kinect depth sensor as direct input methods. Furthermore, the interface consists of freely movable digital items allowing the users to set up dedicated
workspaces and explore datasets on their own. Taking all this into account, the main goal was to provide users, such as students and teachers, a prototype supporting and extending collaborative science learning activities. Therefore, an initial usability study was performed upon these two extended developments (multi-touch and in-air gesture). The initial analysis revealed that the study participants achieved overall better results in the multi-touch scenario compared to the in-air gesture scenario. This led to the conclusion that the in-air scenario is not suitable for complex productive workflows, while the multi-touch interaction is. A distinct advantage in this case could be identified when it comes to the visualization of big amounts of data where multiple users could actively be engaged. All the participants approved the possibilities of the collaborative application and liked the visually driven data visualization and exploration as an offset to traditional workstations with single-user input. This overall finding highlights the need for an integration of new interaction technologies and scenarios in collaborative interactive data visualization, and especially in scenarios related to environmental science learning.

The latest activity we recently carried out with regard to “extensibility” was connected to National Geographic Society’s (NGS) GIS tool, which has been designed to support geographic investigations and encourage collaboration between young citizens and researchers. Our software system was successfully integrated with NGS’ GIS platform called FieldScope. FieldScope has been designed to support geographic explorations and to promote citizen science practices in real-world issues. One of the main drawbacks of NGS FieldScope was the lack of uploading data onsite where the data was actually collected. Thus, the main idea was to extend NGS FieldScope by using our mobile data collection tool. This illustrates the notion of extensibility of our software system in the sense of how sensor data and observations collected using our mobile application combined with the Export function from the Data Aggregation component were used to visualize these data sets in other tool such as NGS FieldScope. The activities we conducted validate the flexibility for data exchange and integration that our software system offers. Our software system has been conceived and based upon the notion of an open and extensible architecture (Vogel, 2012). Figure 9 below depicts the screenshot of NGS’ FieldScope that visualizes water quality data collected using the LETS GO system.
Sustainability of our solution

During the last year of development, our efforts were focused into making the current system a sustainable one, so it can be widely used even after the end of the project. Having in mind the problems we experience with cloud environments (as introduced previously), especially on changes on Google App Engine regarding authentication and limitation of the services, we decided to migrate the LETS GO system (i.e., aggregation server) into our own local environment. The entire environment is based on open source software and open standards. Furthermore, by having full control over this environment we foresee a long maintenance of this software solution and its application across different domains that require mobile data collection and visualization activities. The next step will be making our current solution even more accessible, open and usable for a long period of time.

The user testing study described in earlier sections provided us with additional requirements (Vogel, Kurti, Milrad & Kerren, 2011) that were implemented during the two additional evolutionary prototyping efforts (4th and 5th development iterations, introduced in Table 1). These two additional development efforts in total make five development iterations that made our software system extensible and sustainable while new requirements continuously emerged in these activities. These entire processes made it possible to verify that user requirements were adequately addressed while satisfying their needs. Figure 10 below provides details about our continued evolutionary prototyping efforts (as a continuation from Figure 4 introduced above) across a timeline related to requirements and prototype testing, mapped with the extensibility and sustainability aspects that were considered during these two last iterations. The fourth development stage addressed the extensibility challenges as described above. The fifth development stage made us think to make our system more sustainable towards providing an open platform comprised of a rich set of tools that offer flexible mobile and web based applications that can be deployed by users to support data collection, visualization and collaboration.

The LETS GO system has evolved over a four years period from a prototyping system to become a sustainable one with the possibilities to rapidly be adapted with new features and extensions by taking into consideration the rapid evolution of software and web technologies. In addition, we want to emphasize that the modular design and capabilities of such sustainable system have been conceived with the intention to reduce total platform replacements, where the replacement of a certain component or service as well as the extension of the system with new functionalities becomes increasingly feasible into our solution.

Conclusions and future work

The development lifecycle presented in this paper enabled us to gain valuable insights related to different aspects of mobile and web engineering while developing a system for supporting inquiry based learning activities. During the four years of development efforts, three software prototypes were implemented utilizing service-oriented approaches.
that include mobile, web and interactive visualization modules. The main challenges we identified during these efforts were related to integration, interoperability, extensibility and sustainability of our software system while fostering collaborative science learning activities in the field of TEL. These efforts have been tested with more than 400 users in connection to several trials that took place during this period.

The LETS GO educational activities and tools enabled students to learn in a variety of ways that encompass indoor and outdoor activities across locations and time with the support of sensor technologies, mobile devices and web-based tools. The experiences and knowledge gained during these years enabled us to develop the LETS GO system to a sustainable and robust platform for mobile data collection, visualization and collaboration. The user trials allowed testing the software applications throughout five development iterations on authentic settings, while new requirements continuously emerged in these activities. Reflecting upon our latest development efforts and the results presented in this paper the main findings of our research are discussed in the lines below.

Collaborative technologies, which were used in the iterations of this project, facilitated the adoption of a learner/user-centred approach. The learner/user-centred approach has been suggested by Bonk and Cunningham (1998), where they emphasized “the need to anchor learning into real-world or authentic contexts that make learning meaningful and purposeful”. Sensor kits, mobile devices, web services and interactive technologies nowadays provide us with a vast amount of opportunities of embedding learning activities into real world settings. In these environments the real challenge is the need of matching the dynamic requirements that are generated during learning activities. Hence the main contributions that this paper addresses are the development insights while integrating the technological resources and support for successful implementation of the educational activities related to environmental science learning in authentic settings.

Based on the insights gained from our research efforts we consider that by utilizing an extensive prototyping approach, the discussion of ideas, designs, requirements and implementation possibilities with users/learners the development becomes more easily manageable and understandable. For testing the technical feasibility and understanding whether the technology and implementation behaved as expected, agile development approaches based on prototyping (by combining throwaway and evolutionary approaches) were utilized. They offered an easy and communicative way to test it on real time activities and in authentic settings. Furthermore, this approach helps to find a balance between the design and implementation stages by considering the rapid evolution of technologies.

The integration of diverse heterogeneous device environments where learning activities take place must be based on solutions that promote data exchange, integration and reuse. In our research, we have identified that using open standards technologies for data exchange, promotes systems interoperability and extensibility with new features. Clear cases of such approaches have been the extensibility with interactive technologies and services such as NUI elements and NGS Fieldscope. The initial benefit of this approach seems to be the fact that these new interactions paradigms promote collaboration among users while reflecting upon collected data during outdoors activities.

Moreover, the rapid technological changes affect the flow of learning processes and educational organizations. For enabling rapid changes to be smoothly reflected in everyday activities in this area, there must be well-defined processes to ensure the continual refinement of the applications developed. Facilitating the communication between research projects/researchers and developers on the one hand and research projects/researchers and educators on the other hand, are key factors for the success of these interventions and their sustainability. This approach would enable implemented technologies and applications be closely integrated into everyday educational practices, thus maximizing the benefits in terms of the long-term goals, costs, time, and to satisfy learners/educational institutions with their system. A systematic view on those aspects and their implication for developing sustainable software solutions to support mobile learning could lead to a number of potential benefits:

- Standard based systems
- Constant interaction with users/learners
- Incremental development
- Reduced time and costs
- Expandability
- Flexible change of technologies
- Higher usability
- Easy maintenance and sustainability
In summary, all these identified research insights and benefits were gained during iterative user-centred development cycles of a software system for fostering collaborative science learning activities. Moreover, they provided solid foundations in terms of the possibilities of tackling the requirements for supporting inquiry learning in a flexible manner. From a system perspective, these requirements are best fulfilled by using service-oriented approaches that facilitate interoperability through utilizing open source and open standards and by following the evolutionary approach of prototyping. These findings are directly related to software engineering processes aiming to address the requirements posed by mobile learning scenarios and applications. The issues of interoperability and extensibility of the software solution are directly connected with the possibilities of dynamic reconfiguration of learning spaces to respond to learners’ contextual needs. A sustainable design of technological support to meet these needs it requires to be closely developed and deployed in close iterations with different stakeholders (including also teachers and students). This approach may help teachers and learners to overcome some of the complexity of the learning activities and furthermore it may promote the seamless integration of physical and digital learning resources.

References


