Using Social Software for Teamwork and Collaborative Project Management in Higher Education

Na Li[1], Carsten Ullrich[2], Sandy El Helou[1], Denis Gillet[1]

[1] Ecole Polytechnique Fédérale de Lausanne (EPFL),
1015 Lausanne, Switzerland
{na.li, sandy.elhelou, denis.gillet}@epfl.ch

[2] Shanghai Jiao Tong University,
200030 Shanghai, China
ullrich_c@sjtu.edu.cn

Abstract. This paper discusses the potential role of social software in supporting teamwork and collaborative project management in higher education. Based on the fact that social software has been widely spread among young students nowadays, using it for collaborative learning is believed to increase students' involvement and create learning incentives. Two social software platforms, *Graaasp* and *Google Wave* are examined in terms of sustaining collaborative learning activities. Relevant existing features and possible extensions that enhance the learning experience are addressed. Benefits and challenges resulting from the bottom-up learning paradigm are also presented.

Keywords: Social Software, Collaborative Learning, Project Management, Teamwork, Web 2.0.

1 Introduction

Ever since its emergence, social software such as Facebook, Myspace, Twitter, and LinkedIn has attracted a large number of young users, many of whom have integrated these platforms into their daily life. According to a survey [1], the majority of high school and college students are frequent social software users, who use them actively to stay in touch with friends, share information with each other, and express opinions back and forth. They are called the "Net Generation" [2]. They are highly involved in online social communications and naturally pick up a set of social networking skills like blogging, tagging, rating, commenting, and so on.

Given that social software has already gained its popularity and played an important role in students' daily life, using it for the purpose of collaborative learning is believed to be promising. As indicated in recent studies [3][4][5], application of social software in higher education enhances students' both quantitative and qualitative engagement in learning activities, as well as encourages their active contribution. As a result, the underlying hypothesis in this paper is that, giving students the opportunity to use the same solutions for both social and educational interaction enables to increase learning incentives and reduce the time required to get acquainted with collaborative project management tools.

Although using social software has already become a pervasive phenomenon among students, its potential has not yet been extensively exploited for educational purpose. As indicated in a previous study [6], students do not perceive a connection between their online activities and institutional learning. So, research efforts are needed to illuminate how the Web-based communication and network visualization tools found in today's social software can be incorporated into students' overall learning ecology and their social practices. In this paper, the opportunity of using social software for collaborative learning and project management is investigated through scenarios. In particular, the potential role of social software in supporting informal collaborative learning process in addition to the existing formal ones is discussed. Two social software platforms are especially evaluated with this respect: *Google Wave* (wave.google.com) and *Graaasp* (graaasp.epfl.ch).

The rest of the paper is organized as follows. In section 2, traditional learning management systems are investigated, as well as the existing Web 2.0 social software supporting informal teamwork. Afterwards in section 3, *Graaasp* and *Google Wave* are investigated in terms of supporting collaborative project management. Section 4 provides the scenarios of collaborative project management using social software. Then the benefits and potential challenges resulting from the bottom-up learning paradigm are discussed in section 5. Finally section 6 concludes the paper and addresses future work.

2 Related Work

In traditional learning management systems, team collaboration and project management are conducted within pre-defined structures and modules. As an example, the structure of Moodle (moodle.org) is composed of user, resource, and different modules like course, group, forum, assignment, and quiz [7]. Students can collaborate with each other by group discussions in forums and exchange files by posting attachments. Project planning, coordination and tracking are not supported in Moodle. Another well-known collaborative learning system is BSCW that can be used as both a shared workspace and a document management system [8]. Users are able to exchange documents, contacts, appointments, and tasks within communities. BSCW sustains teamwork organization and project coordination, but it does not account for Web 2.0 features.

In addition to those traditional collaborative learning systems, some attempts have been made to use social software for informal teamwork in the realm of learning. In [9], the opportunities for social software to be used to promote collaborative learning among students are discussed. Existing social services like blog, wiki, social bookmarking, and RSS feeds are integrated to enable teamwork among student groups for common goals. However, documents and group activities are not modeled in their infrastructure. Another notable online learning community is eTwinning [10], which uses TwinSpace as a collaborative workspace for projects. Collaboration is conducted based on blogs, forums, wikis, and file archives. It is a teacher centric system that may impair students' motivation for learning.

In short, most traditional learning management systems adopt top-down schemes, where tutors organize and control the learning processes in pre-defined modules. Students might be less motivated in such a framework as they are given less control of their own learning experience and environment. Although some Web 2.0 social services, which are believed to encourage learning and participation, have been incorporate into learning communities, the application of social software for collaborative learning is still in its early phase and needs to be further explored.

3 Using Social Software for Collaborative Learning

Web 2.0 technologies have brought a new type of participatory medium, which highlights social sharing and active interaction. It is believed to facilitate effective and efficient collaborative learning activities. A handful of social software has already provided a number of Web 2.0 features that can sustain the collaborative learning process and project oriented teamwork. In this section, two social software platforms, *Graaasp* and *Google Wave* are evaluated from this perspective. Existing relevant features that are believed to enhance learning experience will be discussed, while available and missing extensions and plugins useful for collaborative active learning will be addressed as well.

3.1 *Graaasp* and 3A Model

The *Graaasp* social software can be described as a Web 2.0 application that can serve simultaneously as an aggregation, contextualization, discussion, and networking platform, a shared asset repository, as well as an activity management system. It relies on the 3A interaction model [7], which is intended for designing and describing social and collaborative learning environments. The model consists of three main constructs or entities: **Actors** represent entities capable of initiating an event in a collaborative environment, such as regular users or virtual agents. Actors create collaboration spaces where they conduct personal and group **Activities** to reach specific objectives. In each of these activities, actors can take different roles, each of which consisting of a label and an associated set of rights. In addition, Actors produce, edit, share and annotate **Assets** in order to meet activities objectives. Assets can consist of simple text files, RSS feeds, wikis, videos or audio files. In the following part, the crucial *Graaasp* features that facilitate collaborative learning and teamwork are discussed in details.

Collaboration: The design of *Graaasp* follows a bottom-up approach that releases organizational and hierarchical constraints in its structure. Group activities can be constructed either with a flat or with a hierarchical structure. In a flat structured group activity, every member shares equal rights so that no one acts as a supervisor who superintends the learning process. While in a top-down hierarchical structured group activity, the rights of group members vary depending on their different roles. Hence, the supervisor or tutor is able to organize and track the learning process and the project just as what is done in traditional learning management systems. Both tutors and students are entitled to create group activities of any user-defined types like forum, discussion, course, project, and so on. For instance, tutors can create a course activity for organizing a particular course, while students are also able to create a project activity for coordinating their course project. According to different user intentions, the group activities can be either private or public. Private ones are only accessible upon direct invitations, while public ones allow every user to join them. On one hand, group activities could be seen as collaborative workspaces, which enable members to share assets, conduct discussion, and organize their collaborative work by creating sub-activities. On the other hand, group activities could also be considered as tasks in projects. In this case, a project can be planned, coordinated and tracked by managing these tasks.

Roles: In a group activity space, users are allowed to take different roles. As it was mentioned earlier, each role consists of a label and a set of associated rights allowing users to perform diverse actions such as creating or posting new assets in the space, linking or relating items to the space, commenting, rating, tagging, bookmarking, and so on. This is particular useful in collaborative learning environment, for assigning roles to users in the group makes them aware of their duties and gives them the opportunity to concretely collaborate by being allowed to perform specific actions.

Awareness: Another important feature in *Graaasp* is awareness that allows users to be notified of ongoing activities and current status. Users are entitled to subscribe to RSS feeds associated to any 3A entities to trace changes. Additionally, once a user is invited to join a group activity, he/she will get notified by receiving an invitation email.

User Generated Content: As mentioned before, typical Web 2.0 features are supported in *Graaasp*, including tag, rate, comment, link, wiki and bookmark. These features not only enable users to express opinions easily, but also play a crucial role in recommendation of people, learning resources and group activities. Tags allow users to classify their collections in the ways that they find useful, and also facilitate building a folksonomy in the learning community [11]. Tags can be made either public or private, providing the possibilities of both cooperative classification and personal categorization. As far as rates and comments are concerned, they help evaluating the quality of content in open environment. With respect to links, they provide a way to associate relevant items, which makes it easier to discover connections among actors, group activities and assets. In addition, the popularity of wiki provides an opportunity for users to co-create content and collaboratively work towards common goals. Finally, bookmarks help users to organize and share their Web resources in an informal way [12].

Gadgets: *Graaasp* can serve not only as a collaboration platform, but also as a gadget container [13]. Gadgets can run and communicate within *Graaasp*. This feature reinforces the learning experience because it enables useful learning tools to be added and launched during the learning process. Different set of gadgets can be associated to different spaces, making the aggregation contextual. For a given project, users are able to link relevant gadgets to the corresponding space and use them in the specific project context.

Available Extensions in *Graaasp*: In most project management systems, time tracking and milestone management are core features. For the purpose of maintaining the learning process, two microformats are introduced: hCard and hCalendar. Microformats are simple, open design patterns based on existing standards such as Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS) [14]. They are intended to describe common Web content, including people, places, events, reviews and tags, by adding simple markup to human-readable data items. With the help of microformats, semantic information is directly attached to the contents of Web pages [15]. As a HTML adaption of the widely used standardized format including vCard and vCalendar, microformats can be exported to a wide range of external applications like calendars, address books, and so on. As one of the microformats, hCard is utilized to represent people, organizations and places. In *Graaasp*, hCard could be associated with a single person or a group. When a hCard microformat is detected by the browser, users are then able to either download it as a ".vcf" file or directly export it into various contact management applications on their laptops, mobile phones and online Web services. Compared to hCard, hCalendar is a microformat intended to describe events and calendar information including event summary, location, start time, end time, alarm, and so on. hCalendar could be embedded into *Graaasp*, aiming at sustain time tracking and process management. It can serve as a common calendar in a project, a to-do manager for a user, as well as a deadline reminder for an asset. By subscribing to these online calendars, users can get notified of upcoming events continuously.

3.2 *Google Wave*

Google Wave is a Web application that allows the collaborative creation of documents. The documents, called Waves, can contain formatted text, pictures, videos, and other content. Users can reply and edit anywhere in a Wave. A playback function allows to see how the document changes over time and to restore an earlier version. Relevant features in *Google Wave* that enhances collaborative learning and teamwork are presented in details in this section.

Collaboration/Roles: The creator of a Wave can invite other users to participate. Users can either have full access or view access (a read-only access that does not allow editing the content of the Wave). Waves can also be made public and be accessible to all Wave users. Group management is currently done via *Google Groups*. Adding a *Google Group* to a Wave gives access to all its members.

Structuring Waves: A powerful search function combined together with tags provides a very flexible way of structuring Waves. By themselves, Waves all exist on the same level, without any hierarchy. However, users can add tags to Waves, structuring them in the way they prefer. For instance, after assigning one lecture specific tag (*HCI2010*), one group specific tag (*Group1*), and one task specific tag (*assignments*), the search *"to:jack@googlewave.com tag:HCI2010 tag:assignments"* will list all assignments of the course *HCI2010* involving Jack. Searches can be saved, which provides an easy way to find relevant Waves efficiently. Furthermore, users can also create folders and classify Waves into folders. These folders, however, are private, in contrast to tags, which are visible to all the members of a Wave.

Awareness: *Google Wave* offers several mechanisms for notifying users about new Waves and changes. A global email notification alerts users whenever they are added to a Wave and upon changes of Waves they are a member of. Users can also add robots to a Wave to enable alerts for individual Waves. For instance, several XMPP robots are allowed to receive notifications via XMPP.

User Generated Content: Since a Wave is a document, users can edit existing text and easily add pictures, videos, and links. Additional assets such as maps, and voting applications can be added via gadgets. Moreover, users can also add private comments visible only to themselves and peers they invited.

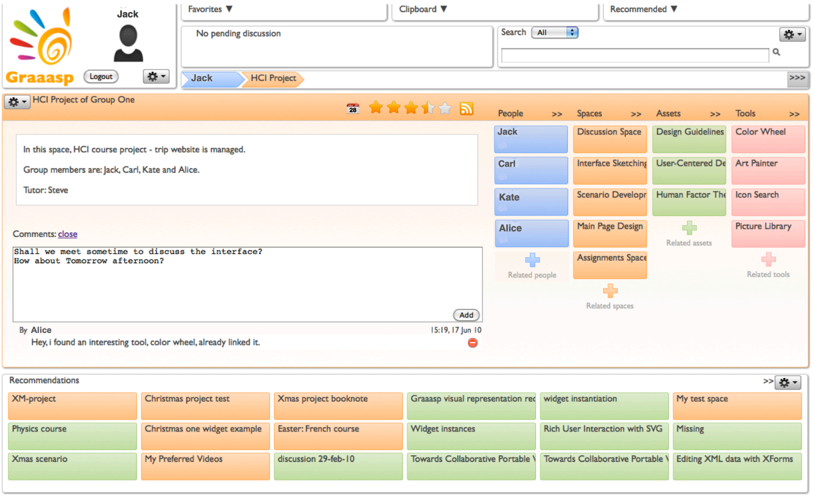
Gadgets and Robots are two powerful frameworks that allow extending the functionality of a Wave. Robots are added to a Wave like a human participant and they are able to interact with a Wave. For instance, a calculator robot replaces the text in the form [1+1] with the solution [1+1=2]. Other robots automatically clean up Waves by removing empty paragraphs and adding missing spaces in punctuation. As another example, the Wolfram Alpha robot answers to the text with the result given by the Wolfram Alpha query engine. Thus "*population China Germany*" would result in a chart comparing the population of China and Germany. After adding a gadget to a Wave, participants can interact with the gadget. For instance, participants can add their locations to a map, vote, draw sketches, and so on. Gadgets can encapsulate complex functionality, as an example, SAP's Gravity gadget enables collaborative business process modeling within *Google Wave*.

4 Scenarios of Collaborative Project Management

To understand the potential role of social software in supporting teamwork and collaborative project management in higher education, a scenario of students coordinating a course project is examined. Steve is the tutor in charge of a project-based course, namely "Human Computer Interaction". During this course, each group of students is asked to accomplish a project and hand in assignments regularly. After a short discussion during a break, four students who are involved in this course, Jack, Carl, Kate and Alice decide to form a group and work together on the project. They agree to design a trip website for the project. In this section, the possible project management process is exploited using both *Graaasp* and *Google Wave*. The interface screenshots for interaction space in *Graaasp* and collaboration Wave in *Google Wave* are illustrated in Fig. 1 and Fig. 2 respectively.

4.1 Scenario Exploitation in *Graaasp*

As presented in Fig. 1, Jack first logs into *Graaasp*, and creates a private activity space called "*HCI Project of Group One*". He then sends out invitations to Carl, Kate and Alice, asking them to join this space with the role of "*Administrator*". This private activity space only allows invited people to become members, so that they are able to carry out their project without being disturbed by others or being observed by the tutor. According to the course instructions and their group agreements, Jack specifies all the milestones and deadlines using the calendar feature in their project space, followed by generating a calendar file accordingly. Four group members subscribe to this shared calendar so that they will get notified whenever there is a deadline.



No pending discussion

In this space, HCI course project - trip website is managed.

Group members are: Jack, Carl, Kate and Alice.

Tutor: Steve

Comments: close

```
Shall we meet sometime to discuss the interface?
How about Tomorrow afternoon?
```

By Alice 15:19, 17 Jun 10

Hey, i found an interesting tool, color wheel, already linked it.

Fig. 1. Screenshot of the Interaction Space in *Graaasp*

In order to look for useful tools for designing the color scheme of their trip website, Alice searches for color scheme gadget in *Graaasp*. She finds an interesting gadget called "*Color Wheel*" and links it to their project space for future use. Also, Carl posts some relevant documents as assets in the space, including "*Design Guidelines for Successful HCI*", "*User-Centered Design*", and "*Human Factor Theory in HCI*".

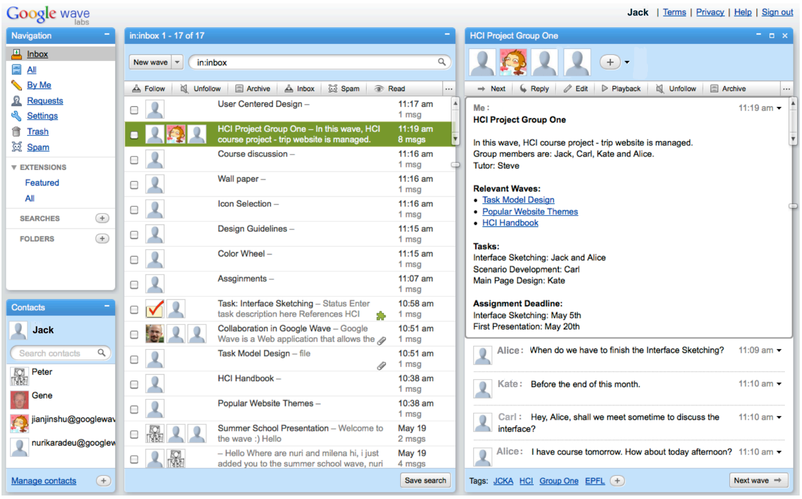
For the purpose of technical discussion, Kate creates a discussion sub-space in the project space. Steve and teaching assistants are invited to join with the role of "*Tutor*" and "*TA*" respectively. Four group members post technical questions and conduct discussion using the space wiki. Steve and teaching assistants subscribe to the RSS feeds of this sub-space so that they can get continuous updates of the discussion in their browser and answer students' questions in time.

Throughout the project process, four group members establish a series of tasks by creating sub-spaces in the project space, such as "*Interface Sketching*", "*Scenario Development*", and "*Main Page Design*". By giving an "*Executor*" role to one or several group members in a particular sub-space, a task is assigned to the responsible people.

For submitting all the course assignments, Jack creates another sub-space called "*Assignments*" in their project space. Steve and teaching assistants are also invited to join this sub-space. Whenever there is an assignment deadline, all the group members are reminded through the shared calendar mentioned before. Prior to the deadline, the responsible student submits the assignment by posting the deliverable as an asset in the "*Assignments*" sub-space. Afterwards, Steve and teaching assistants review the submissions and grade them using the rating feature. Finally, the group members get the grading results either through RSS feeds or through emails, depending on their preference settings.

4.2 Scenario Exploitation in *Google Wave*

In this section, we reuse the scenario setting described above to illustrate the potential role of *Google Wave* in supporting teamwork. Steve, the tutor in charge of the course, sets up the basic environment for the course. He creates a Wave with general information about the course, tagging conventions in order to enable search for course related Waves. He also creates a *Google Group* containing all his students and adds it to the Wave.



User Centered Design – 11:17 am 1 msg
HCI Project Group One – In this wave, HCI course project - trip website is managed. 11:19 am 8 msgs
Course discussion – 11:16 am 1 msg
Wall paper – 11:16 am 1 msg
Icon Selection – 11:16 am 1 msg
Design Guidelines – 11:15 am 1 msg
Color Wheel – 11:15 am 1 msg
Assginments – 11:07 am 1 msg
Task: Interface Sketching – Status Enter task description here References HCI 10:58 am 1 msg
Collaboration in Google Wave – Google Wave is a Web application that allows the 10:51 am 1 msg
Task Model Design – file 10:51 am 1 msg
HCI Handbook – 10:38 am 1 msg
Popular Website Themes – 10:38 am 1 msg
Summer School Presentation – Welcome to the wave :) Hello May 19 2 msgs
– Hello Where are nuri and milena hi, i just added you to the summer school wave, nuri May 19 4 msgs

Me : 11:19 am

HCI Project Group One

In this wave, HCI course project - trip website is managed.
Group members are: Jack, Carl, Kate and Alice.
Tutor: Steve

Relevant Waves:
- Task Model Design
- Popular Website Themes
- HCI Handbook

Tasks:
Interface Sketching: Jack and Alice
Scenario Development: Carl
Main Page Design: Kate

Assignment Deadline:
Interface Sketching: May 5th
First Presentation: May 20th

Alice : When do we have to finish the Interface Sketching? 11:09 am
Kate : Before the end of this month. 11:10 am
Carl : Hey, Alice, shall we meet sometime to discuss the interface? 11:10 am
Alice : I have course tomorrow. How about today afternoon? 11:10 am

Fig. 2. Screenshot of the Collaboration Wave in *Google Wave*

Jack, Carl, Kate and Alice decide to create a Wave for coordinating their project. Based on their group decisions, Jack creates their main group Wave (as shown in Fig. 2), tags it with their group acronym JCKA and invites his peers. Additionally, he links other useful Waves like "*HCI Handbook*", "*Popular Website Themes*", and "*Task Model Design*". Three other group members accept Jack's invitations and discuss their project schedule within the main group Wave. They collaboratively decide on how to structure their assignments using a vote gadget and add a progress-tracking gadget to keep track of their overall work. Then, they create a Wave that contains all the tasks to be done throughout their project. Continuously, they edit the tasks they are working on and add new ones if necessary. Using *Google Wave*'s playback mechanism, they are able to see their progress over weeks.

For every assignment, they create a Wave and collaborate within it. Once they have completed the assignment, they create a copy of the Wave and add Steve as a participant. Steve adds teaching assistants as reviewers to each assignment Wave and includes a rating gadget. The reviewers add private comments visible only to them and Steve, and rate the assignment using the rating gadget.

5 Benefits and Challenges of Bottom-up Learning Paradigm

In traditional formal learning platforms, it is usually tutors who construct the learning environment and lead the learning process. Students are given few opportunities to manage their learning activities according to their own learning intentions. Compared to that, the bottom-up learning paradigm using social software creates a more open and flexible learning environment where structure and hierarchy is not strictly enforced. It brings about both benefits and challenges for the collaborative learning process.

In the first place, the bottom-up learning paradigm allows students to take responsibility of their own learning experience. It not only enables students to create their own workspaces without being observed by tutors, but also encourages them to control their learning process, coordinate teamwork between peers, and manage their projects all by themselves. Such an environment might create students' learning incentives and increase their involvement in learning activities. Furthermore, students of the "Net Generation" have already developed necessary competencies of using social software. Giving them the opportunity of using the same solutions for both personal and educational purposes will somehow reduce the time required to get acquainted with collaborative project management tools. Last but not the least, the open learning environment emphasizes easily opinion expression and thus fosters user-generated content. It facilitates sharing learning resources and accumulating domain knowledge.

However, switching from traditional learning structure to bottom-up learning paradigm also poses several challenges. First, users' active contributions produce a large amount of user-generated content, which may lead to information overflow. Students might lose their focus and find it difficult to select useful learning resources. In this case, providing appropriate evaluation and recommendation is one of the challenges in such an environment. Moreover, given that social software provides an open and agile collaborative learning environment, special attention should be paid to trust issues between tutors and students as well as within the team. Also, the balance between sharing mechanism and privacy management has to be not only tackled, but also well understood by students.

6 Conclusion

Given that a growing number of students nowadays use a variety of social software in their daily life, the opportunity of using social software for collaborative learning is discussed in this paper. In particular, two social software platforms, *Graaasp* and *Google Wave* are investigated, with the intention of supporting collaborative project management and teamwork. Useful features that are believed to enhance the collaborative learning process are addressed, as well as potential extensions.

Additionally, scenarios of social software-based collaborative project management are provided for the purpose of describing how student projects could be conducted using a bottom-up learning paradigm. The usability and acceptability of the bottom-up learning paradigm will be evaluated through user survey in the future. Furthermore, challenges resulting from open learning environment will also be tackled, including information overflow and trust issues.

References

1. Hargittai, E.: Whose Space? Differences Among Users and Non-Users of Social Network Sites. Journal of Computer-Mediated Communication, vol. 13, pp. 276-297 (2007)
2. Tapscott, D.: Grown up Digital: How the Net Generation is Changing Your World. McGraw-Hill, New York (2009)
3. Heiberger, G., Harper, R.: Have You Facebooked Astin Lately? Using Technology to Increase Student Involvement. New Directions for Student Services, vol. 124, pp. 19-35 (2008)
4. King, S., Greidanus, E., Carbonaro, M., et al.: Merging Social Networking Environments and Formal Learning Environments to Support and Facilitate Interprofessional Instruction. Medical Education Online, vol. 14 (2009)
5. Junco, R., Mastrodicasa, J.: Connecting to the Net. Generation: What Higher Education Professionals Need to Know about Today's Students. National Association of Student Personnel Administrators, Washington DC (2007)
6. Greenhow, C., Robelia, B.: Informal Learning and Identity Formation in Online Social Networks. Learning, Media and Technology, vol. 34, pp. 119-140 (2009)
7. El Helou, S., Li, N., Gillet, D.: The 3A Interaction Model: Towards Bridging the Gap between Formal and Informal Learning. In: 3rd International Conference on Advances in Computer-Human Interactions, pp. 178-184 (2009)
8. Bentley, R., Horstmann, T., Trevor, J.: The World Wide Web as Enabling Technology for CSCW: The Case of BSCW. Journal of Computer Supported Cooperative Work, vol. 6, pp. 111–134 (1997)
9. Fitzgerald, R., Barrass, S., Campbell, J., et al.: Digital Learning Communities (DLC): Investigating the Application of Social Software to Support Networked Learning. In: World Conference on Educational Multimedia, Hypermedia and Telecommunications, pp. 2805-2808 (2007)
10. Scimeca, S., Dumitru, P., Durando, M., et al.: European Schoolnet: Enabling School Networking. European Journal of Education, vol. 44, pp. 475-492 (2009)
11. Sen, S., Lam, S.K., Rashid, A.M., et al.: Tagging, Communities, Vocabulary, Evolution. In: 20th Anniversary Conference on Computer Supported Cooperative Work, pp. 181-190 (2006)
12. Farwell, T.M., Waters, R.D.: Exploring the Use of Social Bookmarking Technology in Education: An Analysis of Students' Experiences using a Course-specific Delicious.com Account. Journal of Online Learning and Teaching, vol. 6, pp. 398-408 (2010)
13. Gadgets Specification, http://code.google.com/apis/gadgets/docs/spec.html (retrieved June 2010)
14. Lewis, E.: Microformats Made Simple. New Riders, Berkeley (2009)
15. Mrissa, M., Al-Jabari, M., Thiran P.: Using Microformats to Personalize Web Experience. In: 7th International Workshop on Web-Oriented Software Technologies, pp. 63-68 (2008)