Implications of the Social Web Environment for User Story Education

Terrill Fancott[1], Pankaj Kamthan[1] and Nazlie Shahmir[2]
[1]Concordia University, Montreal, Canada
[2]Shaw Cablesystems G.P., Calgary, Canada
fancott@cse.concordia.ca
kamthan@cse.concordia.ca
nazlie.shahmir@sjrb.ca

Abstract: In recent years, user stories have emerged in academia, as well as industry, as a notable approach for expressing user requirements of interactive software systems that are developed using agile methodologies. There are social aspects inherent to software development, in general, and user stories, in particular. This paper presents directions and means for incorporating the Social Web environment in user story education. In doing so, it proposes a methodology, SW4USE, for such integration. SW4USE consists of a user story process model, USPM, and Social Web technologies/applications that can contribute to the execution of the steps of USPM. A collection of scenarios of use, for both teachers in their classroom lectures and students in their team-based course projects, are presented, and potential learning outcomes are given. The ephemeral and essential challenges in the realization of SW4USE, particularly those related to quality, are highlighted.

Keywords: agile methodology, collaboration, dissemination, process model, user requirement, Web 2.0

1. Introduction

The discipline of software engineering advocates a systematic and disciplined approach towards the development and evolution of software systems. It addresses a number of concerns, including appropriate development of software requirements. The user stories (Cohn 2004) reflect a 'lightweight' approach to user requirements (Maiden, 2008) of interactive software systems, especially those that are developed using certain agile methodologies (Highsmith 2009).

In the past decade or so, software engineering education (SEE) is increasingly being viewed as integral to computer science and engineering education at institutions around the world. In particular, user stories are being introduced in software engineering-related courses at Universities (Bunse, Feldmann and Dörr 2004, Müller 2004) and professional training centers.

In recent years, the Social Web, or as it is more commonly referred to by the pseudonym Web 2.0 (O'Reilly 2005), has emerged as a notable member of the technological environment surrounding SEE, in general, and user story education, in particular. It is interesting to note that, although mutually-independent, agile methodologies and the Social Web have co-evolved.

This paper views education as multi-dimensional and, in that context, aims to highlight the potential applications of the Social Web environment in user story education. In doing so, it rests upon and extends earlier work (Fancott, Kamthan and Shahmir 2011).

The rest of the paper is organized as follows. In Section 2, the necessary background for later discussion is provided and related work is presented. This is followed in Section 3 with the description of SW4USE, a methodology for integrating the Social Web environment in user story education. Next, in Section 4, the implications of SW4USE for teachers and students are provided. In Section 5, directions for future research are outlined. Finally, in Section 6, concluding remarks are given.

2. Background and related work

This section outlines the necessary background on user stories and the Social Web, and situates software engineering students with respect to technology. In doing so, it also highlights related work.

2.1 An overview of user stories

A requirement is an expression of a problem that a (software) system, to be developed, is intended to solve. An agile requirement is a requirement for a software project based on agile principles and practices. In this paper, an agile requirement is considered synonymous with a user story.

Reference this paper as:
Fancott, T, Kamthan, P and Shahmir, N. "Implications of the Social Web Environment for User Story Education"
The Electronic Journal of e-Learning Volume 10 Issue 1 2012, (pp44-59), available online at www.ejel.org

The origin of the notion of a user story can be attributed to Extreme Programming (XP) (Beck 2000). It has since then been adopted by other agile methodologies including Scrum (Schwaber and Beedle 2002) and User-Centered Agile Process (UCAP) (Beyer 2010).

In general, a user story is an informal statement that can be expressed as a structured narrative. This statement includes a *role*, a *goal*, and a specific *value* to the role. For example, consider the following user story statement:

[US-1] A job seeker can search the HRIS for employment opportunities.

Then, 'job seeker' is the role, '(to) search the HRIS' is a goal, and '(to be able to find) employment opportunities' is the value.

A user story statement can be associated with a number of other elements to enrich the description and thereby be useful for different stakeholders. These elements include those related to meta-information. For example, a user story statement is usually associated with estimate and priority so as to be relevant to project managers, and equipped with acceptance criteria so as to be relevant to customers/users.

A *user story book* is a collection of all user stories for a specific software project. It can be expected that a user story book typically consists of several user stories corresponding to different services that an interactive software system offers.

A *user story card* is an index card on which the entire user story is expressed. Initially, paper was used as the sole medium of choice for user story cards. However, as management, in general, and dissemination and presentation, in particular, of a large number of user stories becomes important, electronic 'cards' are increasingly being used.

2.1.1 Significance of user stories

The following characteristics make user stories unique among the different approaches for specifying software requirements:

- **Communication.** The user stories, from elicitation of information from customers/users, emphasize verbal communication. This can lead to an improvement in the relationship between software engineers and customers/users (Alexander and Maiden 2004) from conversing and listening to each other. This, in turn, could eventually contribute to building necessary mutual trust.
- **Comprehension.** The user stories, from the absence of technical terminology in their statements, aim to be comprehensible to customers/users. This is important as the development of software requirements of many interactive systems involves technical as well as non-technical stakeholders.
- **Collaboration.** The user stories encourage collaborative effort in human-centered approaches to the development of interactive software systems. For example, participatory design is such an approach. This is done by engaging customers/users as active participants in the software process.
- **Calculation.** The user stories can be used to make estimates of size and duration of a software project that, in turn, can be used as input for short-term planning (Cohn 2005). It is known that the estimates of size and duration of a software project are useful if they are available as early as possible in the software development life cycle.
- **Conciliation.** The user stories can act as 'bridge' towards reducing the classical chasm between software engineering and human-computer interaction. The disciplines of software engineering and human-computer interaction have essentially evolved independently. The impact of this compartmentalization has not necessarily been positive. In recent years, especially since the increasing acceptance of agile methodologies, it has been realized that a harmonization of the two disciplines is necessary, even mutually beneficial.

2.1.2 Significance of user story education

There are number of reasons for paying special attention to user story education.

The landscape of industrial software development has changed, and continues to change, since the introduction of agile methodologies about a decade ago. It has been shown in a number of surveys that XP and Scrum are currently the two most widely deployed agile methodologies in industry. The user stories are intrinsic to both XP and Scrum.

SEE needs to be mindful of such variations in industry and, to remain relevant, academia needs to reflect and respond accordingly. This position of 'realism' is supported by a number of current approaches to software engineering pedagogy (Navarro and van der Hoek 2009). Indeed, software engineering programs in a number of Universities have been moving in the direction of including user stories as part of their curriculum.

2.2 The new generation of software engineering students

The body of students in software engineering, as in other disciplines, is neither homogeneous, nor static. There are a number of catalysts that have led to behavioral changes (Fogg 2003) in software engineering students over the years, and the one that is especially relevant to this paper is technology.

A *digital native* is a person who was born at the time digital technologies were taking shape and/or has grown up with digital technologies (Palfrey and Gasser 2008). The period in question is 1980 or later. The digital technologies include those that underlie the current non-stationary computing devices, and those that are related to the Internet, in general, and the Web, in particular. The defining characteristics of digital natives (Bernsteiner, Ostermann and Staudinger 2008) include the following: avid users of notebook computers and/or mobile devices, well-connected to each other via distributed computer networks, and rely more on electronic rather than other means of information.

The new generation of software engineering students consists of digital natives. The Social Web is increasingly intrinsic in the daily lives of digital natives. It is likely that the Social Web technologies will become 'transparent' to this new generation of students, and the use of Social Web applications will come naturally. The domination of the Social Web in the daily activities of these students, inevitably, has implications for SEE.

2.2.1 Digital natives, technological determinism, and software engineering education

In education, there are number of arguments against technological determinism, a reductionist theory that presumes technology as the basis for all human activity. In particular, there is objection to assertions such as the following: it is the absence of technology that is the root cause of an educational problem, that the presence of a certain technology will solve an educational problem, and so on.

However, the rise of digital natives may change this traditionally held view. The number of software engineering students that are digital natives is not likely to decrease in the future. Furthermore, at any given time, SEE should be sensitive to the hardware and software technologies being used by the students. It is in the interest of practitioners of SEE to embrace, rather than resist, technological changes, especially if the changes originate from the community they intend to serve. In other words, digital natives may change the perception of technological determinism in education, in general, and SEE, in particular.

2.3 A perspective on the Social Web

The Social Web is the perceived evolution of the Web in a direction that is driven by 'collective intelligence', realized by information technology, and characterized by user participation, openness, and network effects (O'Reilly 2005).

This paper distinguishes between concepts, technologies, and applications related to the Social Web. They can overlap but they are not pairwise synonymous. For example, syndication is a concept, Really Simple Syndication (RSS) is a technology, and an RSS reader is an application.

There are both prospects and concerns associated with a commitment to the Social Web.

2.3.1 Prospects of committing to the Social Web

The following are primary factors that have brought the vision of the Social Web to a mainstream realization:

- **The Enablement of a Many-To-Many Communication Paradigm.** The convergence of social and technical networks has created unprecedented potential for many-to-many communication among people. It has opened new vistas for collaboration and participation. For example, the Social Web, by necessitating collaboration among people with diverse backgrounds, has the potential to reduce the apparent 'compartmentalization' in cultures, including the SEE community.
- **The Broad Availability and Affordability of the Underlying Computing Devices.** The availability and affordability of an appropriate computing device by a person is a necessary condition for accessing the Social Web. In the last decade, there has been an influx of computing devices such as notebook and tablet computers that provide more choice for the consumers in a number of ways. For example, the underlying characteristics of these devices, such as physical size, options for data input/output, screen capabilities, memory, disk space, and processing power, vary considerably. The cost of these devices has also, in general, reduced over time.
- **The Maturation of the Information Technology Infrastructure.** The Social Web can reuse the basic distributed networking infrastructure, including the necessary protocols and addressing schemes, established during the evolution of the Web. The technologies in form of languages for information description, such as markup languages, and information processing, such as scripting languages, have remained relatively stable.
- **The Availability of Technological Implementations as Open Source.** The availability of an open environment in form of open source software (OSS) and open content has played a critical role in the success of the Web, and the same applies to the Social Web. These include, but are not limited to, an increasing number of implementations for LAMP (an abbreviation for Linux, Apache, MySQL, and PHP, each of which is an OSS), Asynchronous JavaScript and XML (AJAX), and Ruby on Rails. This is important for the development of small-to-medium size projects with budgetary constraints. Indeed, there are a number of such projects in the works listed at Go2Web20 (http://www.go2web20.net/).
- **The Awareness, followed by Immense Interest and Significant Participation, by the Public-At-Large.** The Web provides limited opportunities for interaction and participation: the communication is largely unidirectional and users are merely receivers. In contrast, the participatory nature of the Social Web has been a major attraction to the general public. Like electronic mail in the past, there are a number of general-purpose Social Web applications with a relatively low learning curve, which are inviting to those that have minimal technical background. In particular, the Social Web applications such as the Delicious, Facebook, Flickr, Twitter, and YouTube have set the precedence for others to follow.

The aforementioned factors have made students 'first-class' active participants, rather than being mere passive observers, in the use of the Internet for education. They have also helped level the playing field for students by lowering the financial entry barrier.

2.3.2 Concerns of committing to the Social Web

There are a number of ephemeral and essential concerns stemming from a commitment to the Social Web. A subset of these concerns is inherited from the legacy of the Web. The following are some of the essential concerns:

- **Stability.** Many of the current Social Web applications are 'social experiments', available only as Software as a Service (SaaS). These services do not always come with an a priori guarantee of longevity or persistence. For example, a service available at one time may be discontinued at a later time. The discontinuation of the Google Notebook (http://www.google.com/notebook/) is a case in point. The organization that owns a Social Web application could be acquired by another organization, after which the application may significantly change its mission, and therefore the functionality it offers.
- **Multiplicity.** There is currently no single Social Web technology/application that satisfies all, or even most, of educational requirements. This situation is unlikely to improve in the foreseeable future. Therefore, a dedicated commitment to the deployment of the Social Web in SEE, in general, and user story education, in particular, necessitates the use of multiple Social Web

technologies/applications. This, in turn, has non-positive indirections towards course administration and learning curve.

- **Quality.** There are a number of quality attributes of a Social Web application, each relevant to some stakeholder. The three different but related quality attributes, especially relevant to a user of Social Web application, are credibility, usability, and accessibility. The credibility of information is an overarching concern for both Web applications, in general, and Social Web applications, in particular. It becomes especially relevant in education if such applications are being relied upon for teaching and/or learning.

  The ISO/IEC 9126-1:2001 Standard defines usability as "the capability of the software product to be understood, learned, and attractive to the user, when used under specified conditions." Figure 1 shows a subset of attributes of usability, namely understandability, learnability, and operability. The issue of the usability of Social Web applications is especially relevant to this paper as usability is related to learning. The Social Web applications have made some inroads in supporting usability. However, a number of usability-related challenges remain. There is currently no user interface standard for Social Web applications. Indeed, the structure and behavior of Social Web applications for the same domain may vary significantly. These factors do not contribute favorably to learnability.

  The ISO 9241-20:2008 Standard defines accessibility from a usability viewpoint as "the usability of a product, service, environment, or facility by people with the widest range of capabilities." There are unresolved accessibility issues pertaining to Social Web applications. For example, in order to function properly, the current Social Web applications, including the ones covered in this paper, require relatively high-speed Internet connection and the support of latest technologies such as most recent versions of Adobe Flash and ECMAScript.

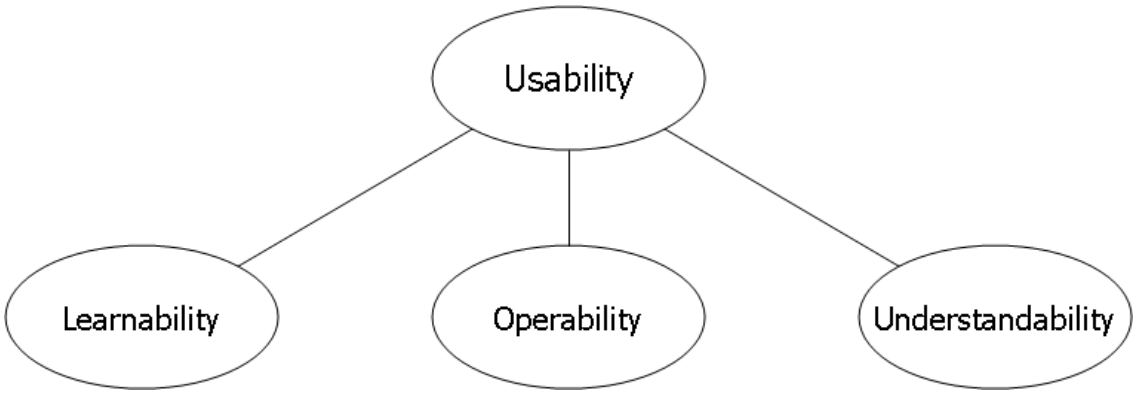

Figure 1: A subset of attributes in a model for usability that have implications towards education

2.4 Related work on software engineering education and the Social Web

To the authors' knowledge, there does not appear to be any substantive work relating user stories and the Social Web. However, there have been a few initiatives towards integrating the Social Web environment in SEE.

The uses of Wiki for teaching software engineering have been reported (Decker et al. 2007, Gotel et al. 2007, Parker and Chao 2007). In particular, it is pointed out that Wikis can be used for publishing and sharing course material, and for managing software process artifacts. However, the correspondence to any teaching strategy or learning theory is unclear.

The usefulness of next generation of social software in engineering education has been demonstrated via the introduction of an application named eLogbook (Gillet et al. 2008). In doing so, the limitations of conventional computer-supported collaborative learning (CSCL) are pointed out. However, the treatment of Social Web technologies is largely one-sided, and eLogbook is not yet mature and its relationship to other social software and technologies is unclear.

Finally, SW4SE2, a methodology for integrating the Social Web environment in SEE, has been proposed (Kamthan 2009). It is concluded that the assortment of technologies and applications underlying the Social Web environment have a number of prospects for SEE. They can provide new opportunities for collaboration between teachers and students, and for collaboration among students;

can create new possibilities for authoring and sharing software project artifacts; and so on. SW4SE2 consists of the following steps: (1) Deciding the Scope of Software Engineering Knowledge, (2) Adopting a Learning Theory and a Teaching Strategy, and (3) Selecting and Applying Suitable Social Web Technologies/Applications to Software Engineering Educational Activities. SW4SE2 has been extended in the direction of different kinds of possible collaborations in SEE (Kamthan 2011).

3. Integrating the Social Web environment in user story education

This section presents SW4USE, a methodology for integrating the Social Web environment in user story education. SW4USE builds on the experience of SW4SE2, and consists of the following sequence of steps:

- Deciding the Scope of User Story Knowledge.
- Adopting a Learning Theory and a Teaching Strategy.
- Selecting and Applying Suitable Social Web Technologies/Applications to the Activities in the Development of User Stories.

3.1 Deciding the scope of user story knowledge

There is support for software requirements in both the *Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering (Software Engineering 2004)* and the *Curriculum Guidelines for Graduate Degree Programs in Software Engineering (Graduate Software Engineering 2009)*. In Software Engineering 2004, as well as in Graduate Software Engineering 2009, there is mention of agile methodologies, but no explicit mention of user stories. There is currently no 'standard', body of knowledge, or reference model for user stories. However, in recent years, user stories have garnered increasing attention, and literature on it continues to grow (Cohn 2004, Cohn 2005, Leffingwell 2011, Monochristou and Vlachopoulou 2007). These previously mentioned references could be used as a basis for user story knowledge for undergraduate and graduate courses.

If the development of user stories is to be effective, it needs to be carried out systematically (Fancott, Kamthan and Shahmir 2011, Kamthan and Shahmir 2010). For that, a user story process model (USPM) is adopted and followed. The mandatory steps of USPM are *Planning*, *Meeting*, *Authoring*, *Reviewing*, and the optional step of USPM is *Publishing*. Figure 2 summarizes USPM. USPM is interspersed, iterative, and incremental. There are a number of activities in each step, details of which are given later.

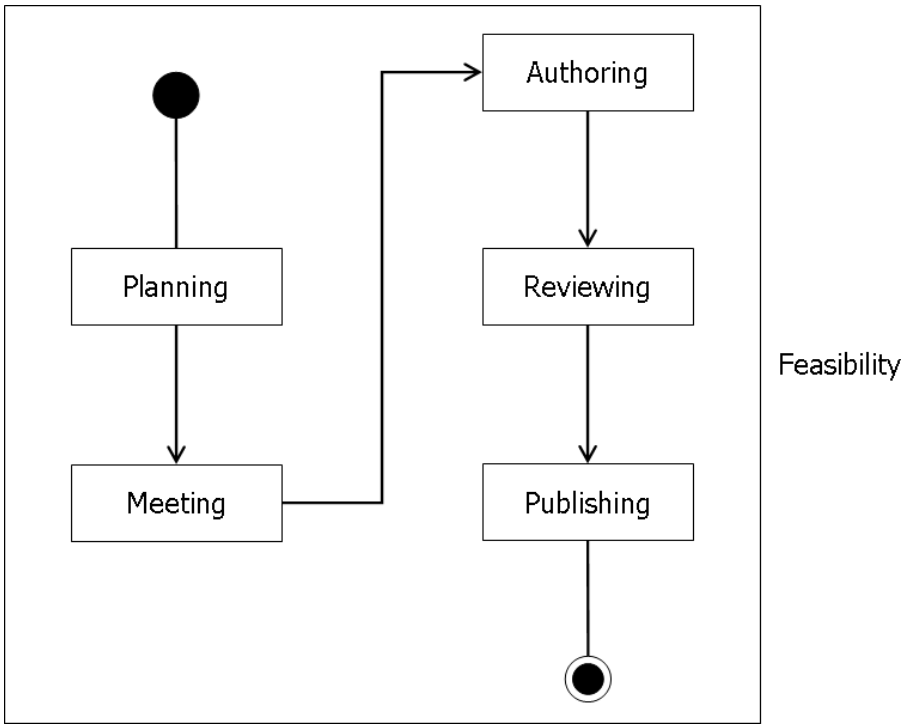

Figure 2: A feasibility-sensitive model for a user story process

There is a need to foster a social environment in the development of user stories at several different levels. A user story is a result of collaboration and negotiation among stakeholders. A user story needs to be shared with and reviewed by stakeholders other than the author. A user story at some point is designed (and implemented); however, if the design is deemed infeasible, then the user story may need to be revised by its author in collaboration with the designer.

3.2 Adopting a learning theory and a teaching strategy

The two theories of learning on which pedagogical strategies are being modeled today are *objectivism* and *constructivism*, each of which has a place in SEE (Hadjerrouit 2005).

For example, a classroom use of Social Web technologies/applications in SEE could be more objectivist than constructivist where the educator plays the role of an 'instructor'. A project use of Social Web technologies/applications in SEE could be more socially constructivist than objectivist where the educator plays the role of a 'guide'.

3.3 Selecting and applying Social Web technologies/applications to the activities in the development of user stories

In software development, the relationships among people and the relationships between people and products (software project artifacts) have conventionally been through separate mediums. The Social Web integrates multiple mediums, and provides a unified platform for these relationships, which are relevant to the development of user stories. Figure 3 illustrates this phenomenon.

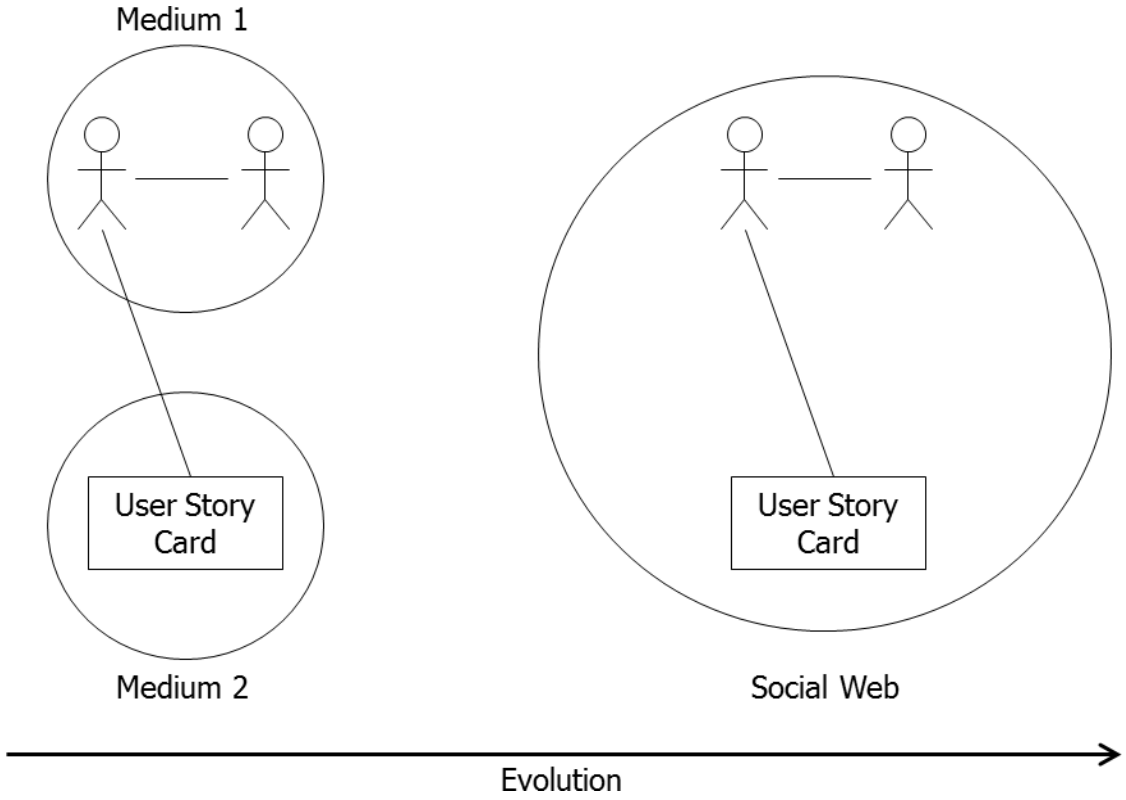

Figure 3: The evolution of mediums for relationships in the development of user stories

The Social Web technologies/applications that are suitable for use both inside and outside the classroom need to be selected. The criteria for selection can include relevancy to the domain (user stories), affordances towards education (undergraduate and/or graduate, as necessary), maturity, cost, and stability. Table 1 lists a sample collection of Social Web technologies/applications relevant to USPM.

The purpose of including an item in the list of Table 1 is awareness, not advocacy or endorsement. There is no claim of completeness of this list. It could be noted that the use of these technologies/applications is subject to license, and these technologies/applications are prone to

evolution in a number of possible directions without any a priori guarantees. A comparison of these technologies/applications is beyond the scope of this paper.

Table 1: A mapping between the set of USPM activities and the set of Social Web technologies/applications

USPM Activity	Example(s) of Social Web Technology/Application
Planning	Doodle, Google Calendar
Meeting	Evernote, Microsoft Office OneNote, UberNote
Authoring	bubbl.us, Google Docs, Wiki
Reviewing	Creately, Gliffy, Google Docs, MockFlow, Wiki
Publishing	Google Docs, Syndication (Atom, RSS), Wiki

4. Implications of SW4USE

This section presents scenarios for both teachers and students, as they relate to the user story education based on the Social Web.

4.1 Implications of SW4USE for teachers

The Web, in general, and the Social Web, in particular, can aid user story pedagogy in a number of ways that may be planned or improvised during the class. To create better understanding of a new topic such as user stories, a teacher may place it in context of other closely related topics such as use cases, or while going through a specific user story statement, a teacher may wish to explore certain terms in that statement. For example, in case of [US-1], the meaning of HRIS (that is, human resource management system) may not be evident to the students. In such cases, resources external to the discussion, such as documents, images, or Uniform Resource Locators (URLs), could be pointed out during the class.

A teacher can also show a part of the dynamism or temporal behavior underlying an agile methodology, in general, and the user story process, in particular, through an animation or a video available elsewhere on the Internet. It is also possible to incorporate external perspectives on a topic by supplementing lectures with video presentations given by others, perhaps by other teachers or book authors. Indeed, there are a number of Social Web applications, such as YouTube, that archive and broadcast dynamic, multimedia information, including video presentations.

There is a variety of information that is generated during a development process. Kanban Board is a tool for visualizing such information from different viewpoints. Figure 4 is an abstraction of a Kanban Board, illustrating the different states in which the user stories are in at a specific point in time.

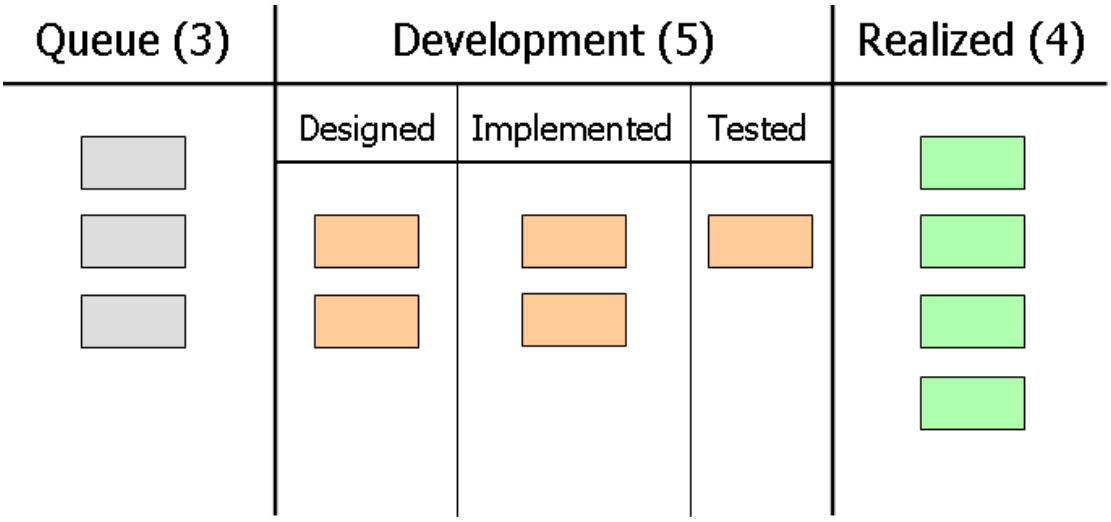

Figure 4: An abstract, simplified Kanban Board illustrating the temporal dynamics of user stories for a given iteration

4.2 Implications of SW4USE for students

There are a number of reasons for including a project component in software engineering courses. The students are supposed to simulate, at least in part, the current state-of-the-practice of software engineering, especially in industrial settings. They are also supposed to develop knowledge and skills involved in solving a non-trivial problem, working both individually and collectively.

This section focuses only on the concerns of user story education that emanate from the realization of software projects in courses. The user stories are especially suitable for software projects that are based on the underlying philosophy, principles, and practices of agile methodologies.

4.2.1 Planning

In planning, there are aspects specific to USPM, the Social Web, and learning.

USPM Aspect: In this step, the resources necessary for authoring user stories, including personnel, means for describing a user story, knowledge on the quality of user stories, and tools for authoring and managing user stories, are identified and selected. The schedules for meeting, authoring, and reviewing are also decided upon.

Social Web Aspect: In setting-up meeting(s) among students, or between students and 'users', accommodating the preferences of each participant can be challenging. The responsibility of finding common date(s) and time(s) often rests solely on the person setting-up the meeting. The logistics of the conventional process can also be unnecessarily time-consuming. The use of Social Web applications such as Doodle (http://www.doodle.com/) and Google Calendar (http://www.google.com/calendar/) can facilitate the arrangement of such meeting(s). These applications have a number of features, including calendar sharing by the participants. Figure 5 illustrates a snapshot of a meeting being scheduled using Doodle to discuss the development of user stories between students S1 and S2, and potential 'user' U1.

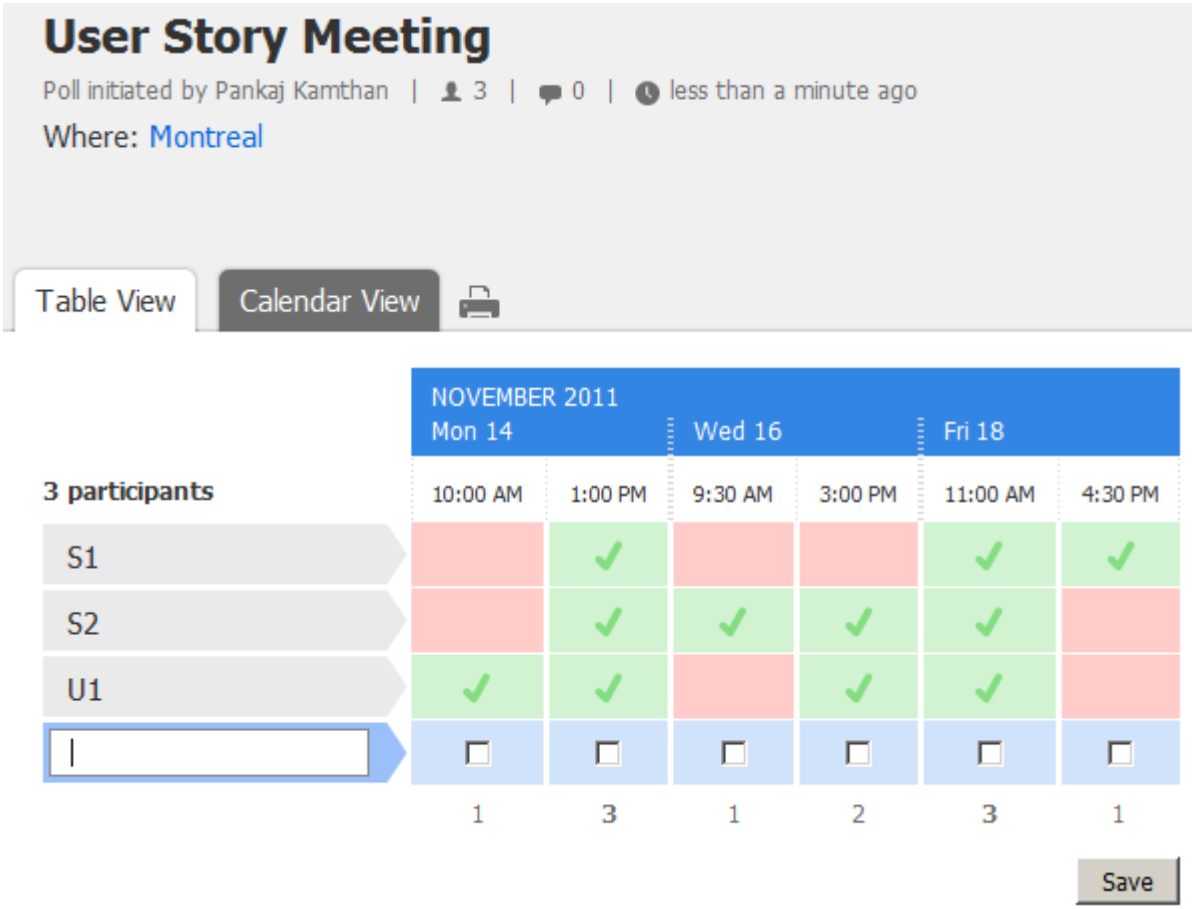

Figure 5: A collaborative environment being used to schedule a meeting to discuss user stories

Learning Aspect: The development of tacit knowledge on planning is important for software project management. The students can learn how to negotiate time and place for meetings, how to organize themselves for a successful meeting, how to estimate and allocate resources, how to develop mutual trust, and so on.

4.2.2 Meeting

In conducting a meeting, there are aspects specific to USPM, the Social Web, and learning.

USPM Aspect: To author the initial set of user stories, a workshop lasting about an hour or two is convened by selected members of the software project team. The 'user' suggests something that he/she thinks that the software system should do. The members of the software project team take those suggestions into consideration in drafting user stories. For the sake of creating a 'team memory' and for reflection, it may be useful to take notes during a meeting and/or record minutes of the meeting. These can be shared with the participants upon conclusion of the meeting.

Social Web Aspect: The activity of taking notes during a meeting is individual. However, it can be useful for the participants to share their notes at the conclusion of a meeting. For taking and sharing notes, there are a number of Social Web applications including Evernote (http://www.bubbl.us/), Microsoft Office OneNote (http://www.microsoft.com/onenote/), and UberNote (http://www.ubernote.com/).

Learning Aspect: The software project team meetings create opportunities for students to learn how to manage time, how to verbally articulate ideas to others (and do so convincingly), how to rebut gracefully, and so on.

4.2.3 Authoring

In authoring user stories, there are aspects specific to USPM, the Social Web, and learning.

USPM Aspect: The software project team selects a specific user model and authors all user stories related to it. The cycle repeats until all user models are exhausted. The user stories are subsequently annotated. In particular, the user stories are identified and prioritized. The result of authoring is a collection of user stories for the current iteration.

Social Web Aspect: There are a number of general-purpose Social Web applications such as Google Docs (http://docs.google.com/) that can assist in authoring user stories.

In collaborative approach to authoring user stories, students often need to engage in brainstorming. There are a number of approaches for brainstorming, and mind mapping is a graphically-oriented approach to realize it.

A *mind map* is a diagram that represents goals, tasks, or other concepts linked to and arranged radially around a central theme or an idea. It is used to generate, visualize, and organize ideas, and as an aid to understanding, problem solving, and decision making. The students can share these mind maps over the Web and, depending on the permissions, read and/or edit others' maps.

For creating and sharing mind maps, there are a number of Social Web applications including bubbl.us (http://www.bubbl.us/) and Mindomo (http://www.mindomo.com/). Figure 6 illustrates a snapshot in time of a mind map, thereby reflecting work-in-progress. In it, four 'stakeholders' (students and 'users'), each represented by a unique color, are in a brainstorming session on the properties of a 'candidate' user story. The 'bubbles' reflect respective inputs by 'stakeholders'.

Learning Aspect: The creation of mind maps encourages thinking, instills creativity, and necessitates collaboration, all of which are an imperative for successful software development. The skills necessary to adequately express software requirements are acquired after a lot of practice (Kovitz 2003), regardless of the language used for stating the requirements. The user stories provide a good entry point for students to learn how to express user requirements in a natural language such as English.

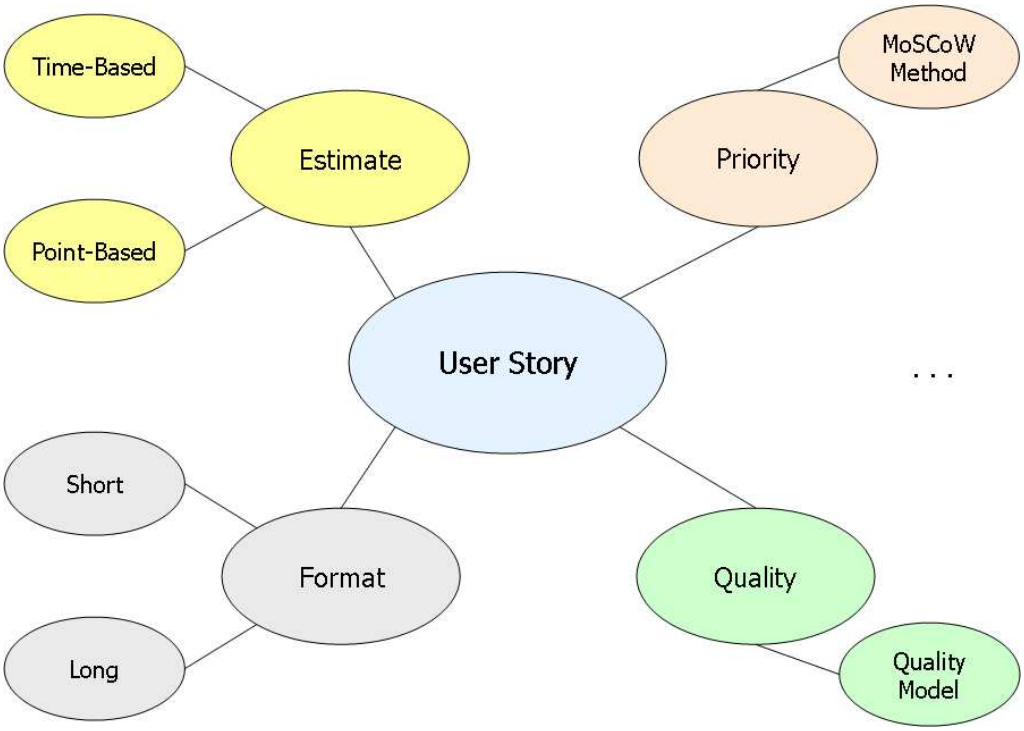

Figure 6: An example of a partial mind map reflecting a brainstorming session on the properties of a user story

4.2.4 Reviewing

In reviewing user stories, there are aspects specific to USPM, the Social Web, and learning.

USPM Aspect: The candidate collection of user stories is prone to evolution for a number of reasons. A user story is reviewed for quality assessment (Wake 2002). If a user story is considered relatively large for the current iteration, it needs to be split; it may need to be reprioritized; and so on.

A user story is also designed (and implemented). However, if the prototype suggests that a user story is deemed inappropriate (say, is unacceptable by a user, or is unimplementable for technical or social reasons), then it must be revised. It is also possible that a user story is rejected after a review.

Social Web Aspect: There are a number of Social Web applications such as Creately (http://creately.com/), Gliffy (http://www.gliffy.com/), and MockFlow (http://www.mockflow.com/) that can assist in collaboratively creating both low- and high-fidelity prototypes. These prototypes can also be shared with others.

Learning Aspect: From participating in a review, the students can learn the need for USPM to be iterative, how to reflect and improve upon their own work, and how to be patient and respectful of each other's work.

4.2.5 Publishing

In publishing user stories, there are aspects specific to USPM, the Social Web, and learning.

USPM Aspect: The user stories need to be published for their subsequent use by others in the software project team. The publication of user stories is also useful for courses that expect team presentations at the conclusion of each major phase of the software development process. The inclusion of meta-information such as unique identifier, author name and contact information, date and time of publication, and version information, become especially relevant in this step.

Social Web Aspect: It is possible to use a Wiki system for user story cards, and use the projects in the Wikimedia Foundation (http://wikimediafoundation.org/) for support as, for example, shown later in Figure 9.

There are several open source flavors of Wiki available today, addressing different target groups and organizational needs. Most flavors of Wiki, including MediaWiki (http://www.mediawiki.org/) and TWiki (http://twiki.org/), can be easily acquired, installed, and administered under commonly-deployed computing platforms. If outsourcing is an option, there are organizations such as Wikispaces (http://www.wikispaces.com/) that provide Wiki-related services and support for educational institutions.

The need to locate relevant information effectively becomes a concern as the number of user stories (and, therefore, the size of user story book) grows. There are different means of finding information on the Social Web, one of which is the use of *folksnomy* (or social tagging). A collection of semantically-related tags, forming a tag cloud, is of special interest. There is support in certain Wiki systems for including tag clouds. Figure 7 illustrates the use of social tagging in general in the context of user stories. The domain-specific terms can serve as source for tags in a tag cloud, and electronic user story cards can serve as destinations.

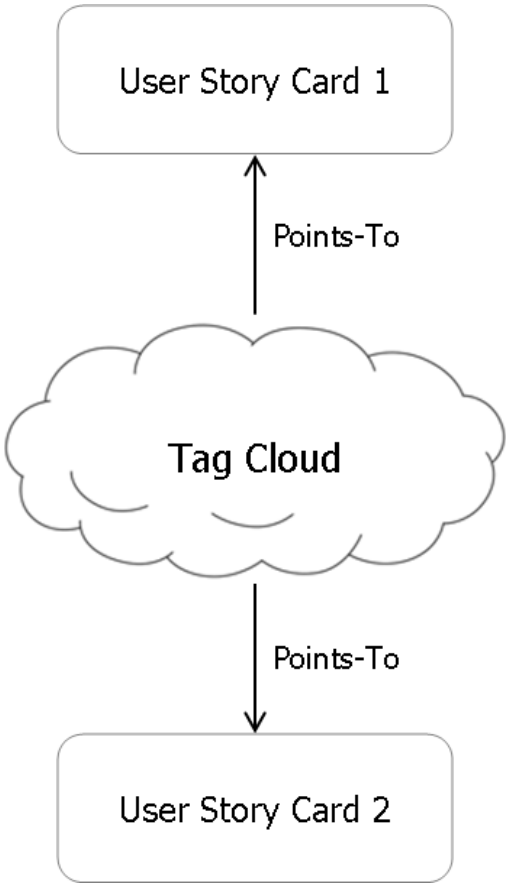

Figure 7: An abstract tag cloud for user story cards

Learning Aspect: The need to publish can help students learn how to communicate and disseminate user stories for the consumption by others. In doing so, they can also learn how to use and manage certain tools needed for the purpose, which may also be useful for their future careers.

4.2.6 Relationships of user stories to other software project artifacts

In general, a user story does not exist in isolation. The students can get an appreciation of the user story ecosystem (Kamthan and Shahmir 2010) by looking at the relationship of a user story to other software project artifacts, specifically, user models and glossary.

User Stories and User Model. There are different types of user models (Junior and Filgueiras, 2005). The ones appropriate for user stories are user role and persona: a user role is abstract and a persona is concrete. The name of a user role constitutes the role name in a user story statement. The goal and value correspond to that of the persona. Figure 8 shows these relationships. The user stories and the user models can reside on the same or different Social Web applications, such as Wiki, in which the user stories can point to user models via the use of hypertext.

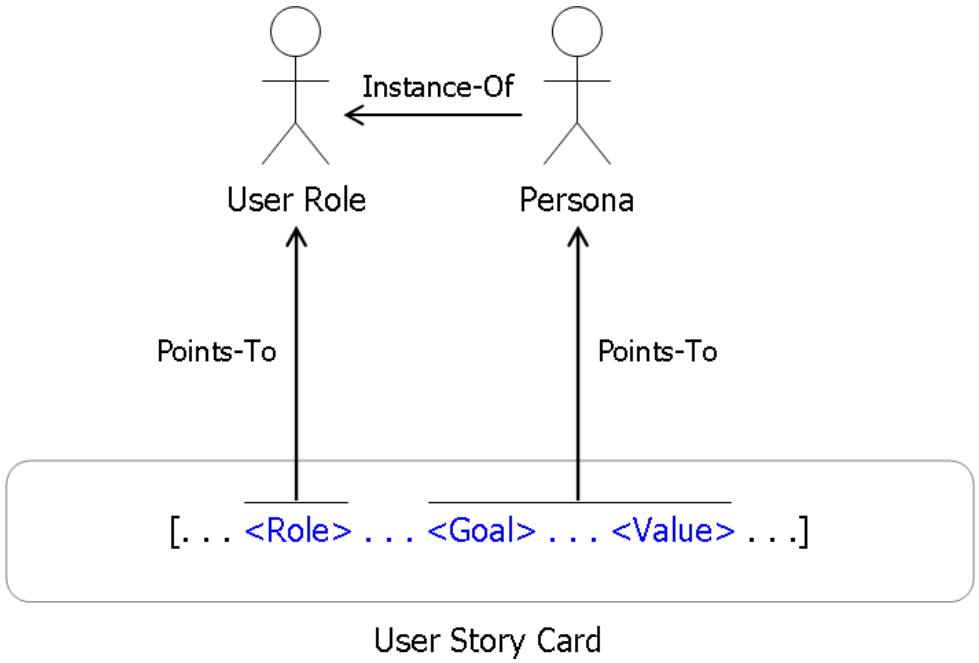

Figure 8: The relationship between a user story card and different types of user models

User Stories and Glossary. It is known that text written in a natural language such as English is prone to misinterpretations by its readers. For example, misinterpretations can arise due to the presence of homonyms, metonyms, neologisms, polysemes, and synonyms. The presence of a software project glossary can help towards reducing the potential for misinterpretation in user stories. For example, as shown in Figure 9, an abbreviation or a domain-specific term in a user story cards could point to the glossary for clarity and disambiguation. This glossary could consist of terms and definitions from publicly available Social Web applications, such as Wiktionary, or other projects in the Wikimedia Foundation.

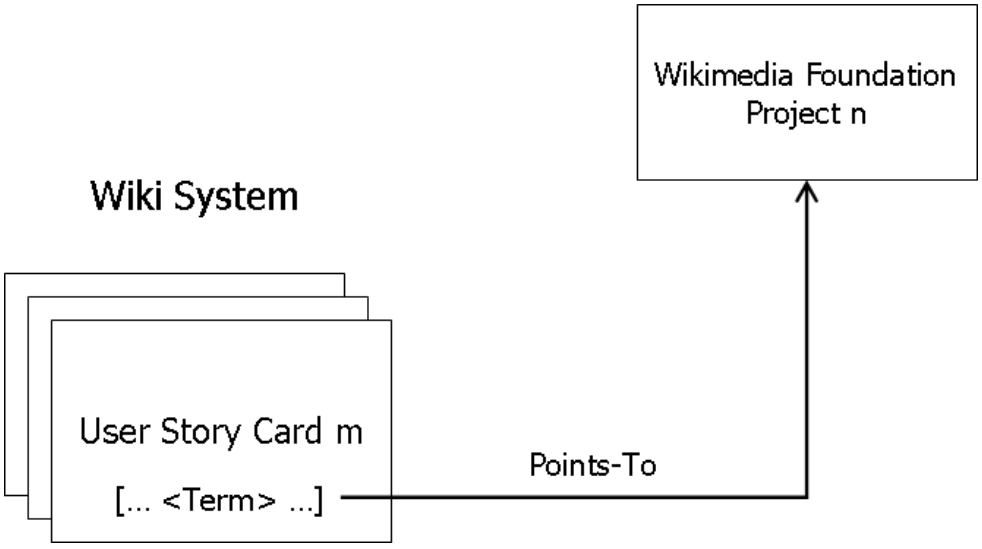

Figure 9: A collection of user story cards residing in a Wiki environment

5. Directions for future research

There are a few anticipated directions in which SW4USE can be extended. In this section, these topics are introduced and discussed briefly.

5.1 Evaluation of SW4USE

Table 1 and the coverage of Section 4.2 imply that the mapping between the set of USPM-related activities and the set of technologies/applications is many-to-many. In such a case, it can be useful to assess the degree of usefulness of each technology/application.

To do that systematically requires commitment to some form of empirical evaluation (Juristo and Moreno 2001). The instruments suitable for such an evaluation in an academic setting include experiments and surveys, although each has its own advantages and disadvantages. The tools suitable for such an evaluation include those being used for learning analytics.

5.2 Reusable experiential knowledge for SW4USE

The reliance on knowledge elicited from experience and described appropriately can be useful in education, as it has been in other disciplines. In recent years, such reusable experiential knowledge has been made explicit in a variety of ways, including pedagogical patterns (Sharp, Manns and Eckstein 2006), in general, and e-learning patterns (Garzotto and Retalis 2009), in particular. A pattern is usually referred to by its name. For example, names of patterns relevant to Section 4 include TEACHING FROM DIFFERENT PERSPECTIVES, STUDENTS DECIDE, and CO-WORK.

If learning theories and teaching strategies form a theoretical basis for education, then patterns contribute towards a practical framework for education. The use of an appropriate collection of patterns could lend a structure to user story education empowered by the Social Web, both inside and outside the classroom. It may also lead to new educational problems, new experiences in solving those problems, and the 'discovery' of new patterns. Therefore, interplay between SW4USE and pedagogical patterns serves as a potential avenue for future research.

5.3 User stories for Social Web applications

The students registered in software engineering programs are uniquely positioned with respect to the Social Web. They can not only be 'consumers' of Social Web applications, but also contribute to the development and/or improvement of these applications.

For example, software engineering students could express the user stories for new Social Web applications or, elicit the user stories for existing applications by means of reverse engineering. In fact, a user story management system (USMS) is a suitable candidate for such a Social Web application.

In other words, as shown in Figure 10, there is a *symbiotic* relationship between user story education and the Social Web, of which one direction is explored in this paper. These possibilities lead to new topics for course projects, but also raise new anthropological, sociological, and technical concerns relevant to SEE. For example, as per Section 2.3.2, understanding and assessing the issue of quality of Social Web applications can serve as a topic of study in a course on software quality or software measurement. Thus, further investigation in this direction is of research interest.

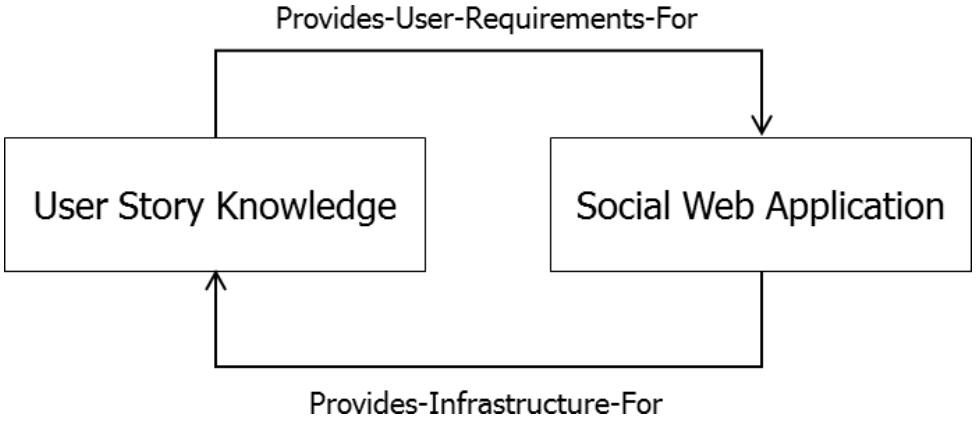

Figure 10: A model for a relationship of mutual benefit between user story education and the Social Web environment

6. Conclusion

The Web has fundamentally changed the practice of SEE, both inside and outside the classroom. Yet another technological revitalization of SEE is inevitable and, the Social Web is a natural candidate for

it. The new generation of software engineering students, and the nature of user story development, provide a suitable platform for exploring the educational possibilities that the Social Web offers.

The Social Web, as this paper illustrates through SW4USE, lends new vistas for user story education. The teachers can use the Social Web for communicating the user story knowledge better and making the sessions more interactive, perhaps even more appealing. The students can use the Social Web for carrying out USPM in their software projects in a manner that is aligned with the anthropological, sociological, and technical aspects of agile methodologies, in general, and user story development, in particular. SW4USE is an initial step, and can be strengthened further by empirical evaluation and incorporation of pedagogical patterns.

It follows from TRIZ, a theory of inventive problem solving, that more of something desirable also brings more of something less desirable, or less of something else also desirable. A commitment to a technology/application has both short-term and long-term impact on cognition and behavior, some of which may be unfavorable and their effects may be irreversible. For example, one such indirection of concern is the potential for dependency. The Social Web is relatively new, and its technological infrastructure is currently in a state of flux. Therefore, a cautious optimism in a commitment of the Social Web environment to SEE, in general, and user story education, in particular, is desirable.

References

Alexander, I. and Maiden, N. (2004) Scenarios, Stories, Use Cases through the Systems Development Life-Cycle. John Wiley and Sons.

Beck, K. (2000) Extreme Programming Explained: Embrace Change. Addison-Wesley.

Bernsteiner, R., Ostermann, H. and Staudinger, R. (2008) "Facilitating E-Learning with Social Software: Attitudes and Usage from the Student's Point of View". International Journal of Web-Based Learning and Teaching Technologies, Vol. 3, No. 3, pp 16-33.

Beyer, H. (2010) User-Centered Agile Methods. Morgan and Claypool.

Bunse, C., Feldmann, R. L. and Dörr, J. (2004) "Agile Methods in Software Engineering Education". The Fifth International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2004), Garmisch-Partenkirchen, Germany, June 6-10, 2004.

Cohn, M. (2004) User Stories Applied: For Agile Software Development. Addison-Wesley.

Cohn, M. (2005) Agile Estimating and Planning. Prentice-Hall.

Decker, B., Ras, E., Rech, J., Jaubert, P. and Rieth, M. (2007) "Wiki-Based Stakeholder Participation in Requirements Engineering". IEEE Software, Vol. 24, No. 2, pp 28-35.

Fancott, T., Kamthan, P. and Shahmir, N. (2011) "Using the Social Web for Teaching and Learning User Stories". The Sixth International Conference on e-Learning (ICEL 2011), Kelowna, Canada, June 27-28, 2011.

Fogg, B. J. (2003). Persuasive Technology: Using Computers to Change What We Think and Do. Morgan Kaufmann Publishers.

Garzotto, F. and Retalis, S. (2009) "A Critical Perspective on Design Patterns for E-Learning". In: Handbook of Research on Learning Design and Learning Objects: Issues, Applications, and Technologies. L. Lockyer, S. Bennett, S. Agostinho and B. Harper (Eds.). IGI Global, pp 113-143.

Gillet, D., El Helou, S., Yu, C. M. and Salzmann, C. (2008) "Turning Web 2.0 Social Software into Versatile Collaborative Learning Solutions". The First International Conference on Advances in Computer-Human Interaction (ACHI 2008), Sainte Luce, France, February 10-15, 2008.

Gotel, O., Kulkarni, V., Neak, L. C. and Scharff, C. (2007) "The Role of Wiki Technology in Student Global Software Development: Are All Students Ready?" Wikis for Software Engineering Workshop (Wikis4SE 2007), Montreal, Canada, October 21, 2007.

Hadjerrouit, S. (2005) "Constructivism as Guiding Philosophy for Software Engineering Education". ACM SIGCSE Bulletin, Vol. 37, No. 4, pp 45-49.

Highsmith, J. (2009) Agile Project Management: Creating Innovative Products. Addison-Wesley.

Junior, P. T. A. and Filgueiras, L. V. L. (2005) "User Modeling with Personas". The 2005 Latin American Conference on Human-Computer Interaction (CLIHC 2005), Cuernavaca, Mexico, October 23-26, 2005.

Juristo, N. and Moreno, A. M. (2001) Basics of Software Engineering Experimentation. Kluwer Academic Publishers.

Kamthan, P. (2009) "A Methodology for Integrating the Social Web Environment in Software Engineering Education". International Journal of Information and Communication Technology Education, Vol. 5, No. 2, pp 21-35.

Kamthan, P. (2011) "An Exploration of the Social Web Environment for Collaborative Software Engineering Education". International Journal of Web-based Learning and Teaching Technologies, Vol. 6, No. 2, pp 18-39.

Kamthan, P. and Shahmir, N. (2010) "Towards an Understanding of the User Story Environment". The Fifteenth IBIMA Conference on Knowledge Management and Innovation: A Business Competitive Edge Perspective, Cairo, Egypt, November 6-7, 2010.

Kovitz, B. (2003) “Hidden Skills that Support Phased and Agile Requirements Engineering”. Requirements Engineering, Vol. 8, No. 2, pp 135-141.
Leffingwell, D. (2011) Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise. Addison-Wesley.
Maiden, N. (2008) “User Requirements and System Requirements”. IEEE Software, Vol. 25, No. 2, pp 90-91.
Monochristou, V. and Vlachopoulou, M. (2007) “Requirements Specification using User Stories”. In: Agile Software Development Quality Assurance. I. G. Stamelos and P. Sfetsos (Eds.). Idea Group, pp 71-89.
Müller, R. A. (2004) “Extreme Programming in a University Project”. The Fifth International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2004), Garmisch-Partenkirchen, Germany, June 6-10, 2004.
Navarro, E. O. and van der Hoek, A. (2009) “On the Role of Learning Theories in Furthering Software Engineering Education”. In: Software Engineering: Effective Teaching and Learning Approaches and Practices. H. J. C. Ellis, S. A. Demurjian and J. Fernando (Eds.). IGI Global, pp 38-59.
O’Reilly, T. (2005) “What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software”. O’Reilly Network, September 30, 2005.
Palfrey, J. and Gasser, U. (2008) Born Digital: Understanding the First Generation of Digital Natives. Basic Books.
Parker, K. R. and Chao, J. T. (2007) “Wiki as a Teaching Tool”. Interdisciplinary Journal of Knowledge and Learning Objects, Vol. 3, pp 57-72.
Sharp, H., Manns, M. L. and Eckstein, J. (2006) “Evolving Pedagogical Patterns: The Work of the Pedagogical Patterns Project”. Computer Science Education, Vol. 16, No. 2, pp 315-330.
Schwaber, K. and Beedle, M. (2002) Agile Software Development with Scrum. Prentice-Hall.
Wake, W. C. (2002) Extreme Programming Explored. Addison-Wesley.

