An Automated Integrated Change and Knowledge Management System

Fangxiao Liu, fzl116@psu.edu
The Pennsylvania State University, University Park, USA

Chimay J. Anumba, anumba@psu.edu
The Pennsylvania State University, University Park, USA

Abdou Karim Jallow, jallow@rowan.edu
Rowan University, Glassboro, USA

Abstract

Change management (CM) and knowledge management (KM) are key aspects of construction project management. Changes can alter service or product variables, and it is challenging to manage the impacts of changes and the associated dependencies. Traditional manual method of managing changes is inefficient and does not fully engage all team members or adequately capture the rationale for changes. In energy efficient retrofits, inadequate management of changes could have negative impacts on building energy performance. In addition, knowledge is often generated as part of the change management process. Given that knowledge has become a fundamental resource for organizations, and can provide organizations with competitive advantage, it is imperative that any new knowledge generated from change management is captured and disseminated. Building on our previous research, which developed an integrated change and knowledge management (ICKM) approach to facilitate the management of changes and knowledge in construction projects, this paper presents the system design and implementation of an automated ICKM. It discusses the requirements and techniques of an integrated change and knowledge management system (ICKMS), the use cases, information exchanges, and the system architecture. The paper also presents the implementation of the system, which included a demonstration of two change examples from ongoing construction projects. The use of the integrated system in construction projects is expected to improve the efficiency and effectiveness of simultaneously managing changes and knowledge. The paper concludes by highlighting the contributions and limitations, and presents recommendations for future research.

Keywords: Change Management, Knowledge Management, Integrated System

1 Introduction

The importance and benefits of knowledge management (KM) and change management (CM) have been increasingly recognized in the Architecture, Engineering and Construction (AEC) industry (Tan et al 2010; Kamara et al 2002), and managing knowledge and changes plays a key role in delivering better performance and services (Jallow et al 2013). The fragmentation and project-based nature of the construction industry hinders the implementation of KM (Forcada et al 2013) and prevents effective capture, reuse and transfer of project knowledge (McCarthy et al., 2000). Throughout the lifecycle of a construction project, changes occur at all stages and a good change management approach is critical to reduce change-related costs and delays (Jallow et al 2013; Liu et al 2014). In the context of energy efficient retrofit projects, challenges of changes and lack of appropriate knowledge could negatively impact energy efficiency goals and lead to a failure to meet the owner's requirements.

An integrated change and knowledge management (ICKM) approach and process, which specifies the process for simultaneously managing changes and knowledge, was developed in previous research (Liu et al 2013) and embedded in construction workflows (Jallow et al 2013). The adoption of ICKM in construction projects will improve efficiency and effectiveness in managing changes and knowledge, and prevent loss of lessons learned in retrofit projects. This paper presents the automation of the ICKM approach. The following sections focus on a review of related work, dependency matrix, system design and architecture. The paper concludes with a brief summary of key findings and future work in this research area.

2 Related Work

Application of IT is a key aspect of project management (Mak 2001). More than 100 project management programs have been introduced in the construction industry, which provide central data repositories for data sharing and facilitate information exchange and communication in a construction project (Burnson 2015).

A number of research projects have been conducted to develop CM and KM systems (Anumba & Pulsifer 2010; Ruikar et al 2007; Motawa et al 2007; King 2007). A knowledge management system (KMS) is a technological platform or infrastructure that is designed to support knowledge sharing and facilitate knowledge exchange processes (Robinson et al 2004). Some systems are similar to Communications and Information Processing (CIP) systems in an organization, but KMS may not be as automated as a CIP system and usually need to involve human participation (King 2007). Knowledge management systems can be categorized into two types: IT tools and non-IT tools (Anumba & Pulsifer 2010). According to Ruikar et al (2007), IT tools (also referred to as technologies) are information technology-based systems for knowledge management; while non-IT tools (also called techniques) support knowledge management without the use of information technologies. Many organizations take advantage of non-IT tools earlier than IT tools (Anumba and Pulsifer 2010). Examples of KMS include CAPRI.NET, OASKMS, CLEVER_KM™, and CBIMKM, (Tan et al 2010; Chong et al 2007; Anumba and Pulsifer 2010; Lin et al 2006).

A change management system (CMS) provides a mechanism for managing changes, including the submittal, review, approval, and notification of change requests (Prosci 2014; Stanford University IT 2014). Ibbs et al (2001) and Arain (2008) analyzed the implementation of change management and proposed some principles for CMS. Table 1 summarizes their key findings. Motawa et al (2006) implemented fuzzy logic-based system for predicting and identifying the risks and impacts of change events by investigating early-stage information. This prediction system can also be used to reduce the disruptive implications of potential changes and determine the likelihood of a specific change event (Motawa et al 2007).

Table 1 Literature on CMS principles

Author	CMS Principles
Ibbs et al., 2001	(1) Promote a balanced change culture; (2) Recognize change; (3) Evaluate change; (4) Implement change; and (5) Continuously improve from lessons learned.
Arain, 2008	(1) Identify variation for promoting a balanced variation culture; (2) Recognize variation, diagnosis of variation; (3) Implement variation, implement controlling strategies; and (4) Learning from past experiences.

3 Dependency Matrix

In order to manage changes effectively in construction projects, it is important to understand and define dependencies (Fewings 2013). A building is a complex system that is composed of dependent elements; therefore, changes may impact more than one element due to interdependencies between building elements.

In order to facilitate dependency checking, a dependency matrix was proposed and developed based on OmniClass Construction Classification System (OCCS). '*Table 21-Elements'* and '*Table 33-Disciplines'* (OmniClass^{TM} 2015), which categorize building elements and their related disciplines respectively, were used to build the matrix. Elements in Level 3 details in '*Table 21-Elements'* were adopted and a list of twelve disciplines was defined according to OmniClass description in '*Table 33-Disciplines'* (as shown in Table 2).

Table 2 Discipline categories

ID	Disciplines	OmniClass Definition	ID	Disciplines	OmniClass Definition
D1	Owner	Real Estate, Facility Owner	**D7**	Masonry Engineer/ Subcontractor	Masonry Contracting
D2	Architect	Design Disciplines, Landscape Architecture, Interior Design, Graphic Design	**D8**	Concrete Engineer/ Subcontractor	Concrete Contracting
D3	Project Manager (General Contractor)	Construction Management	**D9**	Lighting Engineer/ Subcontractor	Lighting Design
D4	Structural Engineer/ Subcontractor	Structural Engineering	**D10**	Fire Protection Engineer/ Subcontractor	Fire Protection Engineering, Fire Protection Contracting
D5	Mechanical Engineer/ Subcontractor	Mechanical Engineering	**D11**	Cost Estimation Coordinator	Cost Estimation
D6	Electrical Engineer/ Subcontractor	Electrical Engineering, Electrical Contracting	**D12**	Financial Analyst	Finance

		Dependent Elements	*Disciplines*	Dependent Elements	*Disciplines*	
		Lighting	*Lighting Design*	**Heating Systems**	*Mechanical Engineering*	
OmniClass #	**Change Proposed Elements**					......
21-02 20 20	**Exterior Windows**	X		X		
		......				

Figure 1 Dependency matrix structure

The dependencies between different elements and the relationships between elements and disciplines were predefined in the matrix according to literature and case studies, so that when a change occurs, the dependent disciplines (marked by 'X') can be identified accordingly. Figure 1 shows the structure of the dependency matrix with some sample data. The dependency checking process includes 3 steps (as shown in Figure 2): (1) when a change occurs, identify the change related elements in the 'change proposed element' column; (2) locate the dependent elements in the 'Dependent Elements' row in the matrix (dependent elements are marked with 'X'); (3) determine the dependent disciplines in the 'Disciplines' categories next to 'Dependent Elements'. For example, in the case of a window glazing change (single glazing changed to double glazing), the change proposed element is *Exterior Windows* (as shown in Figure 1). Then from the matrix we know that the dependent elements are *Lighting* and *Heating Systems*. By identifying the dependent elements, we can then find out the dependent disciplines respectively: *Lighting Engineer* and *Mechanical Engineer*. The dependency matrix was used to automate dependency checking in the system design and embedded in the system codes.

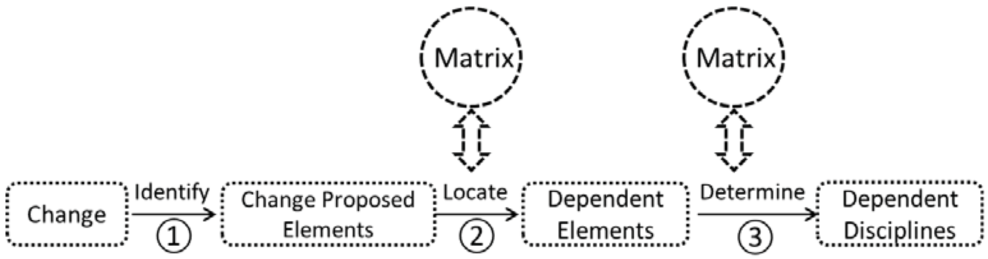

Figure 2 Dependency checking process

4 System Design and Architecture

Based on the ICKM approach, an integrated change and knowledge management system (ICKMS) was developed to support the simultaneous management of changes and knowledge in construction activities.

4.1 Key Features

The implementation of ICKMS in a project setting involves every team member and improves work efficiency for all disciplines. ICKMS has six key features (as shown in Figure 3): (1) undertake dependency checking, (2) track and monitor change request status, (3) adopt escalation feature based on priority and category, (4) notify teams of change information automatically, (5) enable auditability and traceability of change history, and (6) capture lessons learned from changes.

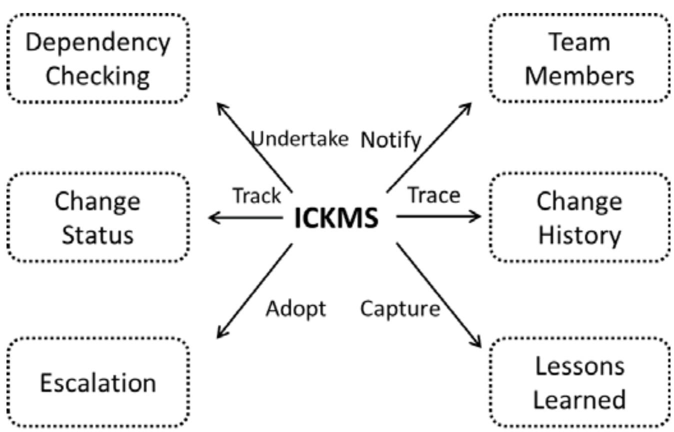

Figure 3 System features

Figure 4 shows the use case diagram of the system that involves the major disciplines (owner, project manager, architects, engineers, general contractor, and sub-contractors) in a construction project.

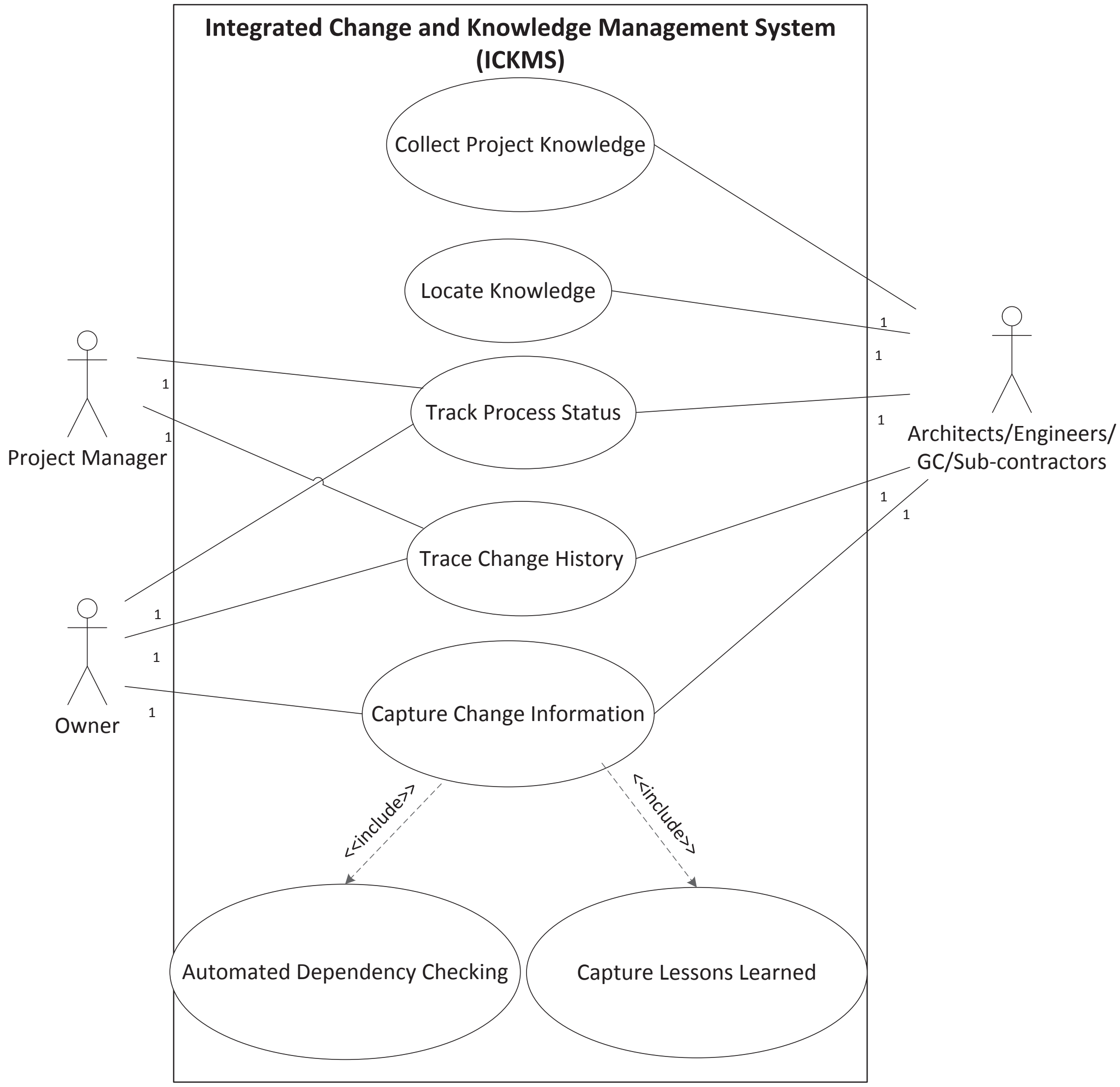

Figure 4 ICKMS Use case

4.2 System Components

ICKMS consists of two core modules: a change management system (CMS) module and a knowledge management system (KMS) module, which are integrated. CMS is designed to manage the change process: identification, evaluation, review (internal and external), confirmation, record, update, implementation and review of change requests. There are four tables in the database: *user accounts* (user profile, username, password and priorities), *change information* (information related to changes), *knowledge repository* (information related to lessons learned and knowledge), and *dependency matrix*. When a request is submitted, a unique ID will be assigned to track all related activities. Table 3 shows the structure of the *change information* table.

Table 3 *Change Information* table structure

Attribute	Type	Description	Attribute	Type	Description
changeID	int(11)	Request ID	**cost**	text	Change related cost
date	datatime	Request date	**ifDelay**	Boolean	Impact of change

project	text	Project name	**num**	int(11)	Number of delays
location	text	Change location	**unit**	text	Unit of delays
reason	text	Change reason	**notes**	text	Supplemental information
description	Text	Change details	**approver**	text	Users to approve change
status	text	Request status	**attachment**	text	Attachment(s) URLs

The CAPRI.NET system was adopted as the KMS component of ICKMS. It is a Web-based system for the live capture and management of project knowledge (Tan et al., 2010). It stores information in a project knowledge file (PKF) so that team members can share knowledge through this platform.

One of the core functions of ICKMS is the capture of lessons learned as potential knowledge from changes. Unlike other existing change management systems that only manage change requests in construction activities, ICKMS is designed to capture lessons learned from users throughout the change management process. The captured lessons are then automatically transmitted and stored in the knowledge repository of the KMS module using Web protocol.

4.3 User Roles

Different disciplines have different requirements for managing change and knowledge. In this study seven roles have been identified for the users of ICKMS, and users are granted specific responsibilities and authorities based on their roles. The system is managed by an 'administrator' that has access to all system features and functions, and is responsible for system maintenance. The 'manager' role is typically undertaken by the project manager and is authorized to access all data stored in the system. Those two roles work for the owner in a project. The 'Subcontractor' role is assigned to users from subcontractors that are permitted to access the functions and data related to their work. These three roles are 'permanent roles' that will be specified when a new account is created and will remain unchanged whether or not the user requests/approves a change request. The other four roles ('requester', 'approver', 'senior approver', and 'performer') are 'temporary roles' and are used in a change request process. The user that submits a request is defined as 'requester', and the people that are responsible for change review/approval are 'approvers'. If a conflict happens and/or approvers fail to respond to a change request in the given time frame, the 'senior approver' will take over the approval role. 'Performers' are people that do the work and implement changes. When a request is approved, they will be notified to proceed the change. All participants involved in a change management process are also responsible for identifying lessons learned. The detailed responsibilities and authorities of each role are listed in Table 4.

Table 4 Roles of user accounts

Role	Category	Responsibility	Authority
Administrator	Permanent	Deploy and maintain the system. Create and manage user accounts.	Access to all system functions and data. Override other users and system rules.
Manager	Permanent	Coordinate other users and manage change requests.	Access to all system functions and data.
Subcontractor	Permanent	Collaborate with other users and respond to manager's requests.	Limited access to system functions and data.
Requester	Temporary	Submit and view a change request. Input lessons learned (if any).	Track request status and view request history.

Approver	Temporary	Review and approve/reject a change request. Input lessons learned (if any).	Approve/partial approve/reject a change request.
Senior Approver	Temporary	Review and approve/reject a change request when conflicts happen and/or approvers fail to respond in a given time frame. Input lessons learned (if any).	Approve/partial approve/reject a change request. Override approvers' decisions.
Performer	Temporary	Implement approved changes. Input lessons learned (if any).	View change request information.

4.4 Development Environment

To ensure cross platform compatibility and accessibility, ICKMS has been designed as a Web-based system. Compared to other traditional applications, a Web-based system is economical to deploy as it only requires applications on a web server and Internet access. I It also provides a reasonably secure environment for users and eliminates the need to monitor client computers, thus reducing the cost and efforts of maintenance (MAGIC 2014).

Apache/MySQL/PHP was used for the development of ICKMS because of the following advantages (Answerguide 2013):

- PHP/APACHE/MYSQL is a solution of open source applications, which are free to download. Both Microsoft-based servers and Unix/Linux-based servers support them;
- Transporting a site from one host to another host is simple as PHP/APACHE/MYSQL is widely supported by Internet hosting providers.

4.5 Website Design

ICKMS is hosted on a web server so that each team member gets access to it. Tables 5 lists the core web pages and corresponding functions.

Table 5 Core web pages and functions

#	Web Pages	Functions
1	My Dashboard	This web page provides graphs of change request data and enables users to view the data they have access to. For examples, users can manage their change requests, track request status and search change history, etc.
2	Advanced Search	Users can define detailed criteria to search the data in the database.
3	Request a New Change	This is the starting page to submit a new change request and input related lessons learned.
4	Review Change Requests	Users can view and approve/deny ongoing change requests and input lessons learned if they are involved in the process.
5	My Change Request Drafts	New change requests are stored here before submission. Users can come back and submit the requests.
6	Internal Messages	Users can communicate with each other through the internal message system.

5 Change Examples and System Operation

Typically, a change request/approval process involves four different user roles: requester, project manager, approvers/senior approvers, and performer. Figure 5 illustrates the workflow to use ICKMS to manage changes and knowledge, as well as the information exchange between users and system database. The first step is that the requester initiates a change request to the ICKMS on the *Request a New Change* web page. The system will then provide recommendations of dependent disciplines (approvers) according to the dependency matrix. Next, the project manager will review the request and dependency recommendations, and manually adjust related approvers if necessary. Then the approvers will be notified and review the change request on the *Review Change Requests* web page. Finally, when the request is approved by all approvers, ICKMS will notify the performer to implement the change. Meanwhile, users will be prompted to input lessons learned when interacting with ICKMS throughout the approval process, and the collected data will be sent to the *Knowledge Repository* table in the database. The *Change Information* table will also be updated in this process to document the status and detailed information of the change request.

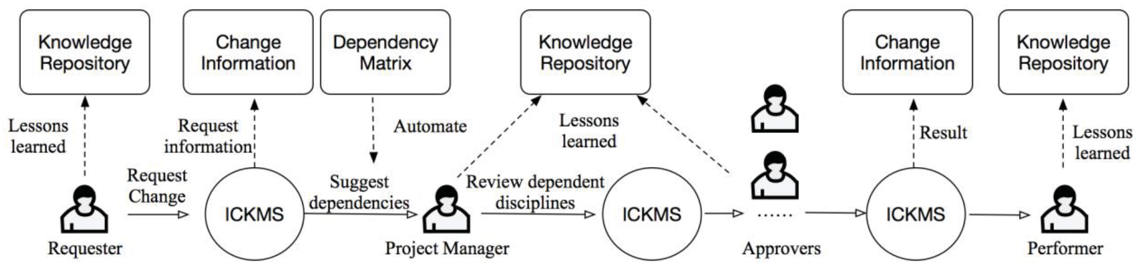

Figure 5 Change approval process

ICKMS plays an important role in this process. It helps integrate the change management and knowledge management processes together and provides a single platform to manage changes and knowledge. It also helps automate the process. When the requester initiates a change request, the project manager will be notified automatically by email. Similarly, the approvers will be notified automatically when the project manager completes the initial review, and the performer will be notified after the change is approved. The system also assigns due day to each user according to the requester's deadline demand. For example, if the requester needs the result within 10 days of submission, the system will divide that 10 days by the number of approvers evenly and assign a due day for each approver. The project manager will then review the due day rules recommended by the system and make adjustments if necessary. If a user fails to respond by the due day, the system will send a second reminder to the user and notify the project manager at the same time.

Two construction projects and their change and knowledge management practices were examined to help test and refine the system. This section focuses on one of the projects, Project A. It involved the renovation of a student residential hall at a major research university. It is an $84 million, design-build project with GMP (Guaranteed Maximum Price) contract, and includes the renovation of four dormitories and recreational commons (RC), and the construction of a new 200-bed dormitory. The project team included 15 subcontractors. Table 6 lists the two change request examples from this project.

Table 6 Change request examples

Location	Cause	Description	Impacted Disciplines
Dorm 1	Requested by owner	Add a sink to the laundry room. The locations of laundry controller, data lines, and two dryer circuit duplex receptacles need to be changed accordingly.	General contractor; architect; owner; mechanical subcontractor;
RC	Unforeseen condition	The existing east steam tunnel foundation was not constructed as shown in the original construction documents. The thickness of the existing footing was approximately ½ of the thickness shown on the original documents.	General contractor; architect; structural engineer; masonry engineer; concrete subcontractor;

Table 7 shows the user account designation for these two scenarios. In comparison, Example 1 requires a simple linear process while Example 2 requires a relatively complicated process. It is important to define appropriate sequence and dependency of approvers involved in a change request, and various testing scenarios can help ensure the system responds to different types of change requests correctly.

Table 7 User account setup

User	Discipline	User	Discipline
User 1	General contractor	**User 5**	Structural engineer
User 2	Architect	**User 6**	Masonry engineer
User 3	Mechanical subcontractor	**User 7**	Concrete subcontractor
User 4	Financial analyst		

In the Dormitory 1 example, the owner requests to add a sink to the laundry room. From the system implemented perspective, the general contractor (user 1) submits the request through the ICKMS on behalf of the owner. The ICKMS will then conduct dependency checking and recommend the involvement of the architect (user 2) and mechanical subcontractor (user 3). Next, the request is reviewed by the user 2 and user 3 in sequence. Finally, the financial analyst (user 4) will make the final decision.
Similarly, for the Recreational Commons example, after the general contractor (user 1) submits the request, the ICKMS will identify the 4 dependent disciplines according to the dependency matrix: architect (user 2), structural engineer (user 5), masonry engineer (user 6) and concrete subcontractor (user 7). The project manager then decides on the sequence to follow for change approval. In this case, user 6 and user 7 can approve the change without each other's input because these two disciplines are independent of each other (as shown in Figure 6).

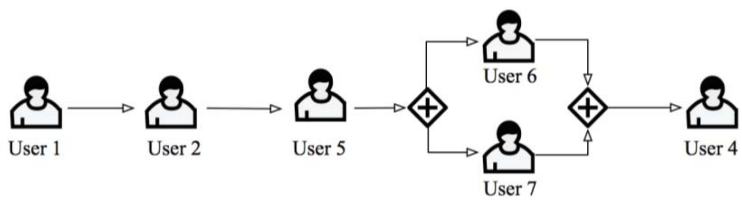

Figure 6 Change approval workflow (example 2)

In both of the above examples, users will also be prompted to input their lessons learned to the system to capture potential change related knowledge throughout the change approval process.

6 Summary and Conclusions

Although some of the existing project management systems in the market are equipped with the functions of managing changes, these systems lack the features to simultaneously manage changes and knowledge, capture change-related lessons learned, and store CM and KM information in a structured data format (unlike ICKMS). ICKMS provides an efficient solution to manage changes and knowledge throughout the lifecycle of a project.

In conclusion, an integrated change and knowledge management system (ICKMS), which is capable of proactively and simultaneously managing changes and knowledge, has been developed and presented. The new system can be used in construction projects to capture lessons learned from changes, and manage and track change request processes. It eliminates the gap between CM and KM activities and prevents loss of potential knowledge in CM activities. The Web-based feature makes ICKMS easy to deploy and maintain and compatible with all types of operating systems. Future work will include evaluation of the functionality of ICKMS using case studies and Satisfiability Modulo Theories (SMT) based model checking.

References

Answerguide. (2013) Which is better: PHP Apache and MySQL or ASP IIS and MsSQL?. *<http://wiki.answers.com/Q/Which_is_better_PHP_Apache_and_MySQL_or_ASP_IIS_and_MsSQL>* (March 2013).

Anumba, C. & Pulsifer, D. (2010) Knowledge management systems for construction. *Proc. of Construction Research Congress 2010*. Banff, Alberta, Canada, May 8th-10th 2010. pp. 687-696.

Arain, F. (2008) IT-based approach for effective management of project changes: A change management system (CMS). *Advanced Engineering Informatics.* 22(4). pp. 457-472.

Burnson, F. (2015) Compare Construction Project Management Software. *Software Advice.* <*http://www.softwareadvice.com/construction/project-management-software-comparison/*> (June 2015).

Chong, N., Uden, L. & Naaranoja, M. (2007) Knowledge management system for construction projects in Finland. *International Journal of Knowledge Management Studies*. 1(3/4). pp. 240-260.

Fewings, P. (2013) *Construction project management: an integrated approach*. Routledge.

Forcada, N., Fuertes, A., Gangolells, M., Casals, M. & Macarulla, M. (2013) Knowledge management perceptions in construction and design companies. *Automation in Construction.* 29. pp. 83-91.

Ibbs, C., Wong, C. & Kwak, Y. (2011) Project change management system. *Journal of Management in Engineering*. 17(3). pp. 159-165.

Jallow A., Liu F. & Anumba C. (2013) The need for change and dependency management in energy efficient retrofit projects. *Proc. of the 2013 Architectural Engineering Conference - Building Solutions for Architectural Engineering*. State College, PA, USA, April 2013. pp. 133-144.

Jallow A., Liu F., Lee S., Anumba C.J. & Messner J. (2013) An approach to integrate change and knowledge management in construction workflows. *Proc. of the 30th CIB W78 International Conference*, Beijing, China, October 8th-12th 2013. pp. 788-797.

Kamara, J., Augenbroe, G., Anumba, C. & Carrillo, P. (2002) Knowledge management in the architecture, engineering and construction industry. *Construction Innovation: Information, Process, Management*. 2(1). pp. 53–67.

King, W. (2007) Keynote Paper: knowledge management: a systems perspective. International *Journal of Business and Systems Research*. 1. pp. 5–28.

Lin, Y., Wang, L. & Tserng, H. (2006) Knowledge exchange through web map-based knowledge management system in construction: Lessons learned in Taiwan. Automation in Construction. 15(6). pp. 693-705.

Liu F., Jallow A. K, Anumba C. & Messner J. (2013) Integration of change and knowledge management processes in energy efficient retrofit projects. *Proc. of the 2013 ASCE International Workshop on Computing in Civil Engineering*. Los Angeles, CA, USA, June 2013. pp. 226-233.

Liu F., Jallow A. K., Anumba C. J. & Wu D. (2014) An integrated framework for embedding change management in Building Information Modeling. *Proc. of the 2014 International Conference on Computing in Civil and Building Engineering*. Orlando, FL, USA, June 23rd-25th 2014. pp. 439-446.

MAGIC. (2014) The benefits of web-based applications. <*http://www.magicwebsolutions.co.uk/blog/the-benefits-of-web-based-applications.htm*> (December 2014).

Mak S. (2001). A model of information management for construction using information technology. *Automation in Construction*. 10(2). January 2001. pp. 257-263.

McCarthy, T., Kahn, H., Elhag, T., Williams, A., Milburn, R. & Patel, M. (2000) Knowledge management in the designer/constructor interface. *Proc. of the 8th International Conference on Computing in Civil and Building Engineering*. Reston, VA, USA. 2000. pp. 836-843.

Motawa, I., Anumba, C. & El-Hamalawi, A. (2006) A fuzzy system for evaluating the risk of change in construction projects. *Advances in Engineering Software*. 37(9). pp. 583–591.

Motawa, I., Anumba, C., Lee, S. & Peña-Mora, F. (2007) An integrated system for change management in construction. *Automation in Construction*. 16(3). pp. 368–377.

OmniClass™ (2015) <*http://www.omniclass.org/*> (June 2015).

Prosci. (2014) Change management: the systems and tools for managing change. <*http://www.change-management.com/tutorial-change-process-detailed.htm*> (November 2014).

Robinson, H., Carrillo, P., Anumba, C. & Al-Ghassani, A. (2004) Developing a business case for knowledge management: the IMPaKT approach. *Construction Management and Economics*. 22. pp. 733–743.

Ruikar, K., Anumba, C. & Egbu, C. (2007) Integrated use of technologies and techniques for construction knowledge management. *Knowledge Management Research & Practice*. 5. pp. 297-311.

Stanford University IT. Change management system. <*https://itservices.stanford.edu/service/changemgt*> (November 2014).

Tan, H., Anumba, C., Carrillo, P., Bouchlaghem, D., Kamara, J. & Udeaja, C. (2010) *Capture and reuse of project knowledge in construction*. John Wiley and Sons.