www.itcon.org - Journal of Information Technology in Construction - ISSN 1874-4753

INTEGRATED CHANGE AND KNOWLEDGE MANAGEMENT SYSTEM - DEVELOPMENT AND EVALUATION

SUBMITTED: August 2018
REVISED: March 2019
PUBLISHED: March 2019 at https://www.itcon.org/2019/7
EDITOR: Amor R.

Fangxiao Liu, Ph.D., Assistant Professor
Department of Sustainable Technology and the Built Environment, Appalachian State University,
Katherine Harper Hall Room 20, 397 Rivers Street, Boone, NC 28608, USA;
email: liuf@appstate.edu

Chimay J. Anumba, Ph.D., D.Sc., Dr.h.c., FASCE, Dean and Professor,
College of Design, Construction and Planning, University of Florida,
1480 Inner Road, Gainesville, FL 32611, USA;
email: anumba@ufl.edu

Abdou Karim Jallow, Ph.D., Associate Professor
School of Business, Slippery Rock University,
1 Morrow Way, Slippery Rock, PA 16057, USA;
email: abdou.jallow@sru.edu

Patricia Carrillo, Ph.D., FCIOB, CEng, FICE, Associate Dean and Professor
School of Architecture, Building and Civil Engineering, Loughborough University,
Ashby Road, Loughborough, Leicestershire LE11 3TU, UK;
email: p.m.carrillo@lboro.ac.uk

SUMMARY: *Knowledge is often generated as part of the change and dependency management (CDM) process. Given that knowledge has become a fundamental resource for organizations, it is imperative that any new knowledge generated from CDM is captured and disseminated. In energy retrofits, inadequate management of changes and dependencies could have negative impacts on building energy performance. This paper presents the development and evaluation of an Integrated Change and Knowledge Management System (ICKMS) in this scope. It discusses the use case, requirements, information exchanges, and the system architecture. ICKMS can be used in projects to manage changes and dependencies, track change histories, and capture lessons learned from changes. The proposed ICKMS eliminates the gap between CDM and knowledge management (KM) activities, and prevents loss of potential knowledge in CDM processes. The use of the integrated system in construction projects is expected to improve the efficiency and effectiveness of simultaneously managing changes, dependencies and knowledge.*

KEYWORDS: *change and dependency management, knowledge management, integrated system*

REFERENCE: *Fangxiao Liu, Chimay J. Anumba, Abdou Karim Jallow, Patricia Carrillo (2019). Integrated change and knowledge management system - development and evaluation. Journal of Information Technology in Construction (ITcon), Vol. 24, pg. 112-128, http://www.itcon.org/2019/7*

1. INTRODUCTION

The application of information technology (IT) is a key aspect of project management (Mak, 2001). More than 100 project management programs have been introduced in the Architecture, Engineering and Construction (AEC) Industry, which provide central data repositories for data sharing and facilitate information exchange and communication in a construction project (Burnson, 2015).

Managing knowledge and changes plays a key role in delivering better performance and services (Tan et al., 2010, Jallow et al., 2013b). 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., 2013a, Liu et al., 2014). The fragmentation and project-based nature of the construction industry also hinders the implementation of KM (Forcada et al., 2013), and prevents effective capture, reuse and transfer of project knowledge (McCarthy et al., 2000, Jallow et al., 2011). The need for tools to support construction project management has been widely recognized in construction industries, and "companies recognize that insufficient control of changes plays a major role in accidents" (CCPS, 2008, pp. 2). In energy retrofits, inadequate management of changes, dependencies and knowledge can lead to critical problems such as inefficient project management and poor energy performance, and could result in a failure to meet the owner's requirements and energy efficiency goals (Liu et al., 2013).

2. KEY ISSUESS IN KNOWLEDGE AND CHANGE MANAGEMENT

2.1 Management of Knowledge, Changes and Dependencies

According to Stewart and Ruckdeschel (1998), knowledge has become the most important factor and resource for organizations today. It is dynamic "justified true belief" created by individual and organization interactions, and answers the "how questions" (Nonaka et al., 2000, Ackoff, 1989). Knowledge is created through cognitive efforts and contains judgment compared to data and information (Tuomi, 1999). In the scope of this research, knowledge refers to 'the possession of validated lessons learned that originate from the project development and operations, as well as the ability to locate their sources' (Liu et al., 2013). According to Kok et al. (2003), knowledge management is "a systematic and organized approach to improve the organization's ability to mobilize knowledge to enhance performance". Knowledge management (KM) contains a series of procedures to manage an organization's current knowledge processing pattern and improve its outcomes (Elgobbi, 2010, Davenport and Prusak, 1998).

Changes generate iterative cycles and uncertainties in construction, and they can occur at various times in all phases (Motawa et al., 2007, Lee and Peña-Mora, 2007). According to Liu et al. (2009), a fault in a system may lead to new faults to other components due to direct and indirect dependencies among the system components. Therefore, management of dependencies is critical in locating faulty states. Good change and dependency management (CDM) enables the evaluation of associated effects, as well as the capture and dissemination of change history.

When a change is requested, it is often managed through an iterative process, during which a large amount of information is processed and new knowledge may be generated. For example, a new lesson learned could be the energy performances for different window glazing types. The architect may propose the use of the single-glazed window type, but the engineer could suggest a change to the double-glazed window type due to a 10% energy consumption reduction based on simulations. However, such lessons learned are not documented in a well-established mechanism, so they may not be efficiently captured or used for future projects. As a result, this gap causes loss of knowledge and makes it difficult for knowledge reuse in future change evaluations and resolutions. Hence there is a need for a system to collate these changes, identify dependencies and retain this knowledge.

2.2 Change and Knowledge Management Systems

A change management system (CMS) provides a mechanism for managing changes, including the submittal, review, approval, and notification of change requests (Prosci, 2014). Ibbs et al. (2001) and Arain (2008) analyzed the implementation of change management and proposed core principles for CMS. Motawa et al. (2007) implemented a fuzzy logic-based system for predicting and identifying the risks and impacts of change events by investigating early-stage information.

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). With an increasing number of vendors providing such a variety of IT tools, different organizations have adopted different knowledge management solutions (Anumba and Pulsifer, 2010). Examples of KMS include Capri.net, OASKMS, CLEVER_KM, and CBIMKM. Capri.net is a Web-based system to facilitate live capture and sharing of knowledge (Tan et al., 2010). OASKMS (Open Application Sharing Knowledge Management System) is a knowledge acquisition and communication application developed for Vaasa city construction department (Chong et al., 2010). CLEVER_KM is a commercialized knowledge management strategy formulation and problem definition tool for projects and organizations (Anumba et al., 2005). The CBIMKM (Construction BIM-based knowledge Management) system was developed in a 3D design environment to help engineers better reuse project knowledge (Lin, 2014).

Although a number of knowledge management systems have been developed in previous research, none of them focus on the management of change-related knowledge and/or are able to facilitate capture and reuse of lessons learned in CDM processes.

2.3 Integrated Change and Knowledge Management (ICKM) Approach

An Integrated Change and Knowledge Management (ICKM) approach was developed to address the fragmentation of CDM and KM processes (Liu, 2015). In the lifecycle of a project, lessons learned from CDM processes (e.g. change requests, change approval, and change implementation, etc.) will be captured and integrated with the KM processes. In addition, previous knowledge can be retrieved and reused from the KM processes to facilitate the CDM processes. The ICKM approach established an integrative mechanism to systematically manage changes, dependencies and knowledge, disseminate knowledge among different disciplines, and support information exchange in both CDM and KM activities. Its implementation in energy retrofits will facilitate the management of change and knowledge issues, and contribute to meeting the energy efficiency goals.

As part of the ICKM approach, a dependency matrix was developed based on OmniClass Construction Classification System (OCCS) to automate the dependency checking process (Liu, 2015). It works as a set of rules that are predefined using IF-THEN format and can be embedded in a system database. Although dependency relationships can be retrieved from the matrix, it is important to manually review the result and make adjustments, if needed. The purpose of the matrix is to give recommendations on dependencies and facilitate decision-making processes.

3. SYSTEM DESIGN AND DEVELOPMENT

3.1 Key Features

A good system prototype should have a process flavor for simultaneous management of changes, dependencies and knowledge while construction activities are performed. This will require automated system functionality (dependency checking and email notifications), and a Web-based platform to support efficient collaborations and easy access. Functions such as information exchange, process coordination and integration between systems should also be included. In addition, to implement the integration, the system must address the necessary features including knowledge capture, storage and distribution (Tan et al., 2010). Therefore, the Integrated Change and Knowledge Management System (ICKMS) was designed with the following key features:

- Automate dependency checking: A dependency matrix was embedded into the system prototype. When a new change is requested, ICKMS will be able to provide recommendations on affected disciplines and approval sequences.
- Track change status: A change request typically needs approvals from multiple approvers before the final result. ICKMS is able to track the progress of each change request and provide updates to users.
- Notify team members: When a change request is submitted/reviewed, the affected disciplines will be notified automatically via emails.
- Trace change history: Change information and history are stored in the system database. Users can also search for previous change requests in ICKMS.

- Capture lessons learned: Lessons learned will be captured simultaneously in the CDM process. When a user submits a new change request, they will be able to input new lessons learned (if any). For example, an energy engineer may recommend a change to the window type from single-glazed to double-glazed based on simulations. The engineer can input the lesson learned about the energy performance of double-glazed windows in energy retrofit projects (for example, 10% less energy consumption than the single-glazed window type). This will help manage changes and knowledge in an integrative way, and prevent loss of knowledge.

3.2 System Design

Every member in a project team may have the need to request new changes and the opportunities to generate new lessons learned. The ICKMS prototype is designed for all disciplines involved in a construction project, including owners, project managers, architects, engineers, general contractors and subcontractors, etc. There is also a need for a system administrator that is responsible for system deployment and maintenance.

3.2.1 Use Case

The discipline configuration is important in identifying dependencies and predefining the dependency rules. Fig. 1 shows the use case diagram of ICKMS that involves the major disciplines (owner, project manager, architects, engineers, general contractor, and sub-contractors) in an energy retrofit project.

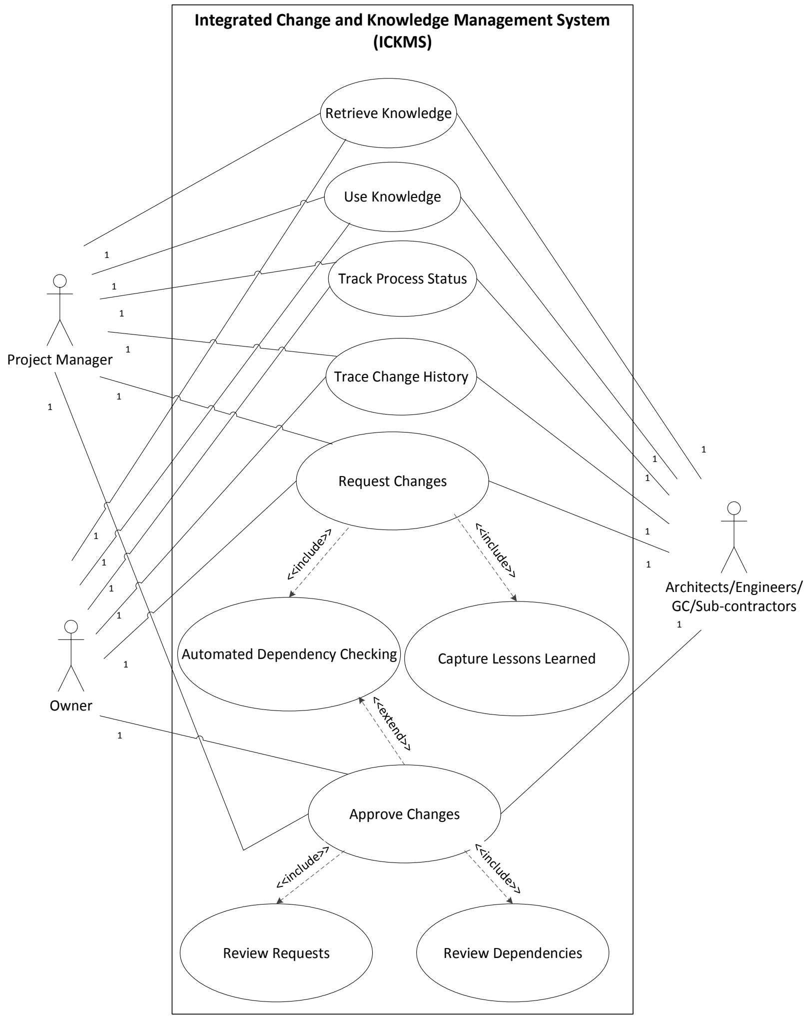

FIG. 1: Use Case Diagram.

Disciplines are categorized according to OmniClass Construction Classification System (OmniClass™, 2015). A unique ID is assigned to each discipline to identify different user roles (as shown in Table. 1).

TABLE. 1: 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	Client Representative	Finance

3.2.2 User Roles

Different disciplines have different requirements for managing changes, dependencies and knowledge. In this research, seven roles have been identified (as shown in Table. 2), and users are granted specific responsibilities and authorities based on their roles. The system is managed by an *administrator* who is responsible for system maintenance and has access to all system functions. The manager role is typically undertaken by the project manager who is authorized to access all data stored in the system. 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. They will remain unchanged whether or not a user is involved in a change approval process. The other four roles (*initiator, approver, senior approver, and performer*) are 'temporary roles', and will be only used in a change request process. Users who submit new requests are defined as the *initiators*, and users who are responsible for change reviews/approvals are *approvers*. If a conflict occurs and/or an approver fails to respond to new requests in the given time frame, a *senior approver* will take over the approval role. *Performers* are people that do the work and implement changes, and will be notified when a change is approved. All participants involved in a change approval process are also responsible for identifying lessons learned.

TABLE. 2: 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.
Initiator	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.

3.3 System Components

ICKMS is composed of two main sub-systems: knowledge management system (KMS), and change and dependency management system (CDMS). Each sub-system has its own database. A knowledge repository is the database in the KMS that supports the KM process and stores lessons learned. The change repository is the database in CDMS that stores information of on-going change requests and previous change histories.

Unlike other existing change management systems that only manage change requests, ICKMS is designed to capture lessons learned from users from CDM processes. The Capri.net system (Tan et al., 2010), a Web-based system for the live capture and management of project knowledge, was adopted as the KM component of ICKMS. Capri.net is capable of storing knowledge generated from experience or lessons learned by project teams, as well as knowledge from other sources (for example, knowledge generated from annotations on drawings during design reviews).

CDMS is designed for the CDM process including identification, evaluation, internal/external review, confirmation, record, update and implementation of change requests (Liu et al., 2013). Over the past few years, information exchange has received great attention requiring integration between applications to facilitate interoperability. In this approach, interoperability matters immensely to enable information exchange.

3.3.1 Change and Dependency Management System (CDMS)

As one of the sub-systems in ICKMS, CDMS was designed to facilitate CDM activities. Its functions include: (i) Submit a new change request; (ii) View change information, including track status of pending requests and search the database for previous change history; (iii) Review change requests and make decisions (approve, partial approve, or reject); (iv) Send out notifications automatically to related team members (users) when a new change is requested or a new status is updated in a request; (v) Automate the dependency checking process by the pre-defined dependency matrix to facilitate the management process; and (vi) Capture change related lessons learned from users and communicate with KMS (Capri.net). Table. 3 lists the core web pages and their corresponding functions.

TABLE. 3: Core Web Pages and Functions of CDMS.

#	Web Pages	Functions
1	Request a New Change	This is the starting page to submit a new change request and input related lessons learned.
2	Advanced Search	Users can define criteria to search the database.
3	My Change Requests	This web page displays a summary of a user's previous requests.
4	Review Change Request	Users can view and approve/deny ongoing change requests.
5	Review Change Request (admin)	This is the administration page for the project manager only. It is used to conduct initial review of every change request before the request is sent out for approval. Project manager is also able to manually adjust approval sequences and involve users based on system's recommendations.

In terms of the CDMS database design, the following information is needed to effectively manage changes and dependencies: users and their disciplines; change requests and status; and the dependency matrix. Therefore, five tables were developed:

- *UserInfo* (user account information);
- *Disciplines* (discipline set up information)
- *ChangeRequest* (change request information
- *ChangeProgress* (status of each change request: changeProgresID, changed, sequenceNew, etc.); and
- *Dependency* (pre-defined dependency matrix).

When a request is submitted, a unique ID ("changeID" in the *ChangeRequest* table) will be assigned to track all related activities. The Dependency and *ChangeRequest* tables are linked via attribute 'elementID' so that the system can query the dependency matrix information and provide information regarding approvers based on the input for 'elementID'. Table *ChangeRequest* and *ChangeProgress* are linked via attribute 'changeID' so that each change request submission has its information stored in *ChangeRequest* table and approval progress stored in *ChangeProgress* table. Table *Dependency* and *Disciplines* are linked via attribute 'disciplineID' to track the dependency information. Table *UserInfo* is linked with *Disciplines* (via 'disciplineID' attribute) and *ChangeRequest* (via 'userID' attribute) so that initiators of change requests and users from dependent disciplines can be identified.

3.3.2 Integration with Capri.net

The integration of CDMS and Capri.net was achieved by information exchange using HTTP/PHP protocol. The system architecture is shown in Fig. 2. A new attribute "KnowledgeID" was added in the *ChangeRequest* table in CDMS's database. It is used to link CDMS's database and Capri.net database so that each change request record is associated with its knowledge record. The Web-based interface is the front-end application where users interact with the system. A middle tier-Web Server is used to link the client side with the database server (back-end). Data is transmitted to/from the back-end side via Open Database Connectivity (ODBC) API using SQL. Change/dependency related data will be processed in the CDMS database (MySQL) and lessons learned/ will be processed in the KMS database (Microsoft Access). The following information will be collected for each lesson learned and sent to Capri.net (Tan et al., 2010): *project title, knowledge category, knowledge type, knowledge topic, details, condition for reuse, and captured from*. The main data schema of the ICKMS is shown in Fig. 3.

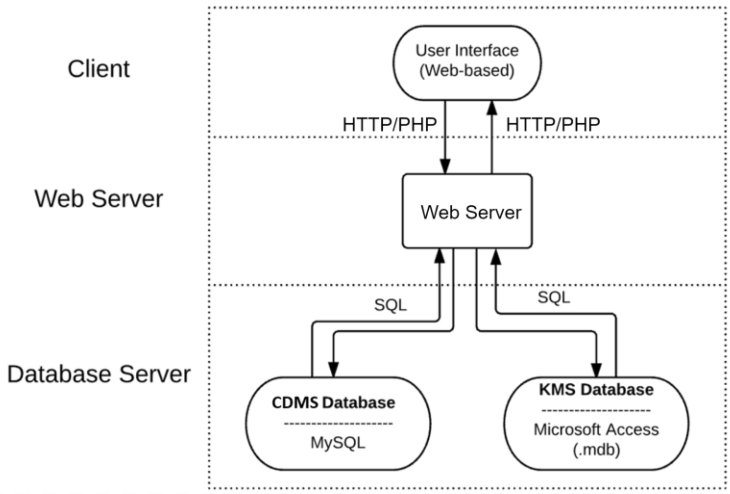

FIG.2: System Architecture.

Previous knowledge can be retrieved by a user through the Google™-like search functions or the 'Advanced Search' function in Capri.net system (Tan et al., 2010). By interacting with the Capri.net interface, users will be able to reuse knowledge in current or future projects. The development of Capri.net is not within the scope of this paper and more details can be found in the work of Tan et al. (2010).

3.4 ICKM Process

Typically, a change request/approval process involves four different user roles: initiator; project manager; approvers/senior approvers; and performer. Fig. 4 shows the ICKMS workflow, as well as information exchange between users and system databases. The first step is that the initiator submits a change request to the ICKMS on the Request a New Change page. If the user has new lessons learned, he/she can also input that information on the same page. The system will then provide dependent disciplines (approvers) according to the dependency matrix, and transmit lessons learned (if any) to the Capri.net database. Next, the project manager will review the request and dependencies, and manually adjust related approvers if necessary. This step ensures that the change request is valid and has all required information. It also grants the project manager the authority to adjust approval sequence (dependency). Then the approvers will be notified and review the change request on the Review Change Request page. Finally, when the request is approved by all required approvers, ICKMS will notify the performer to implement the change. The CDMS database (*ChangeRequest* table and *ChangeProgress* table) will also be updated in this process to document the details and status of the change request. Meanwhile, previous knowledge can be accessed from Capri.net database to support the change approval process.

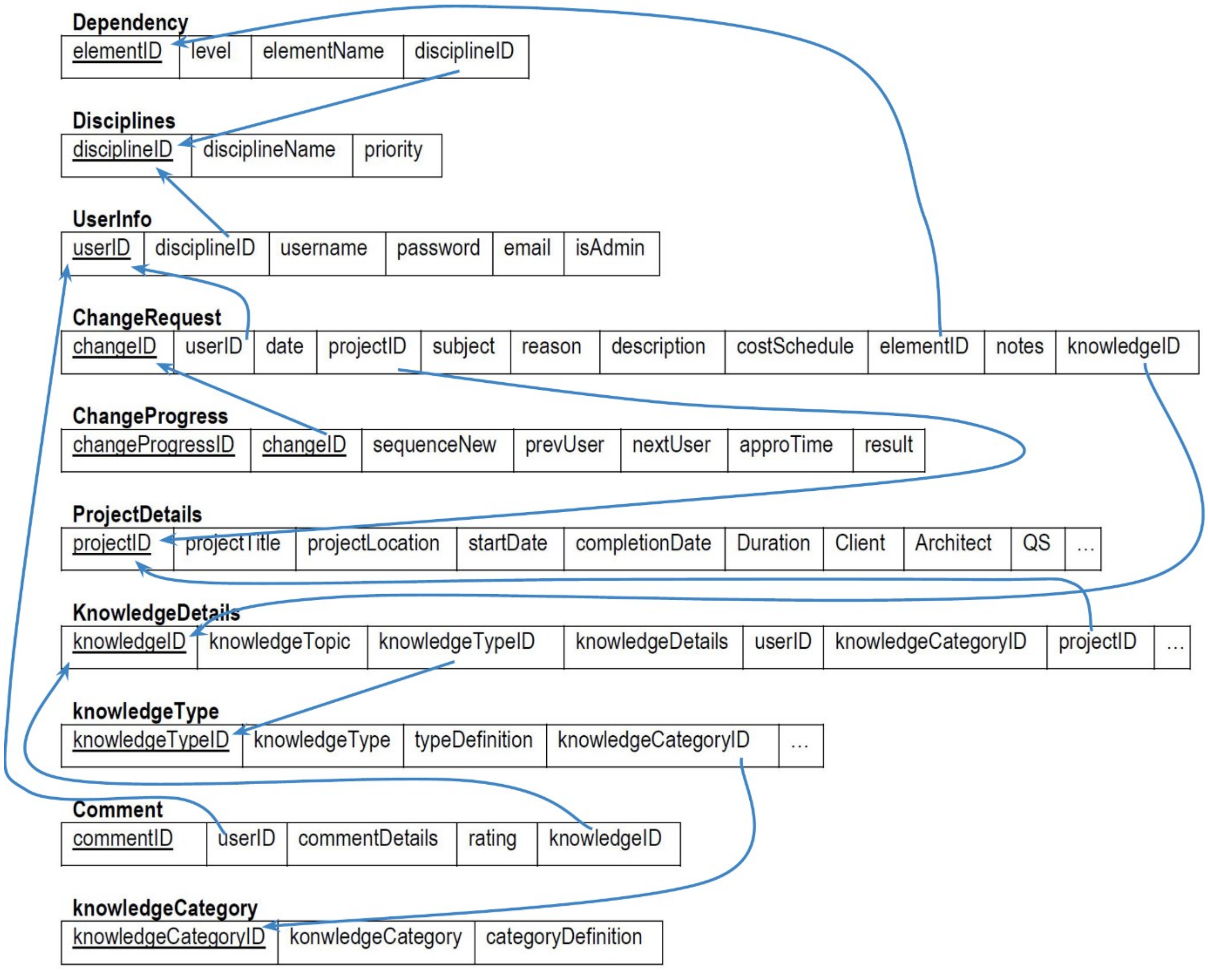

FIG. 3: Main System Data Schema

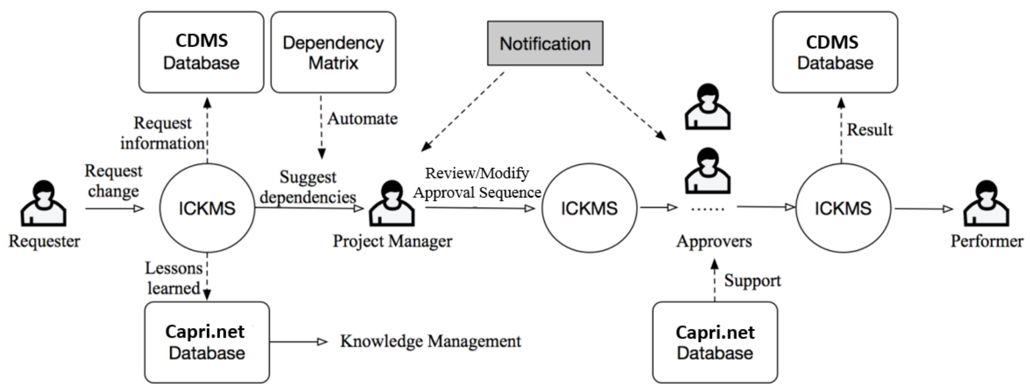

FIG.4: ICKM Workflow.

3.5 User Interface

As an example, Fig. 5 shows the screenshot of the Request a New Change page. Users can submit change requests and lessons learned on this page. It has two parts: a top part to collect change request information and a bottom part to collect lessons learned. In this prototype, the information that the system collects for a new change request includes: project; subject; description; reason; cost & schedule; notes; and affected building element. Users will be required to choose the affected building element that is related to the change from the dropdown list. This information is necessary for the ICKMS to make dependency recommendations. The input from the top part is stored and managed in the CDMS database, and the input from the bottom part is transmitted to the database in Capri.net (knowledge repository) using HTTP/PHP protocol (as shown in Fig. 2). Project managers can review dependencies and manage approval sequences (as shown in Fig. 6), and approvers can approve/deny change requests (as shown in Fig. 7).

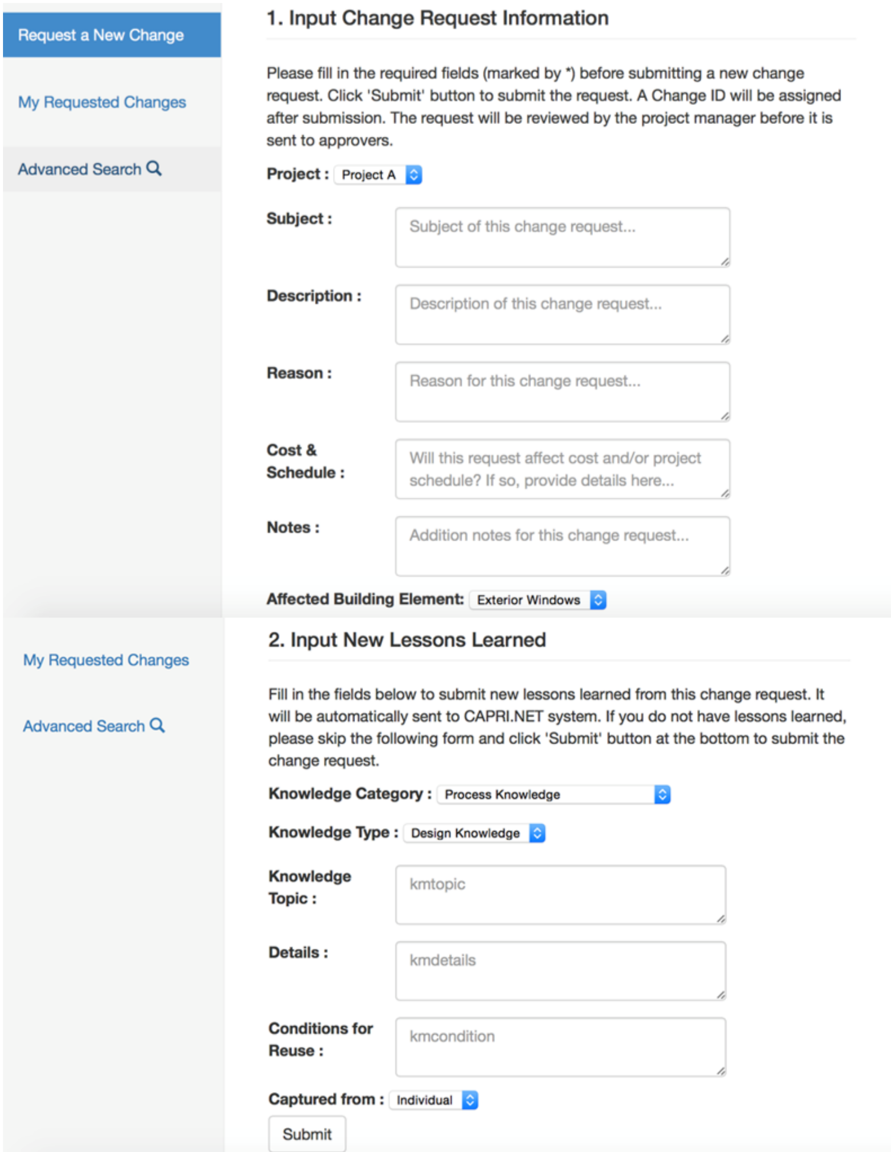

FIG. 5: User interface example - Request a New Change page.

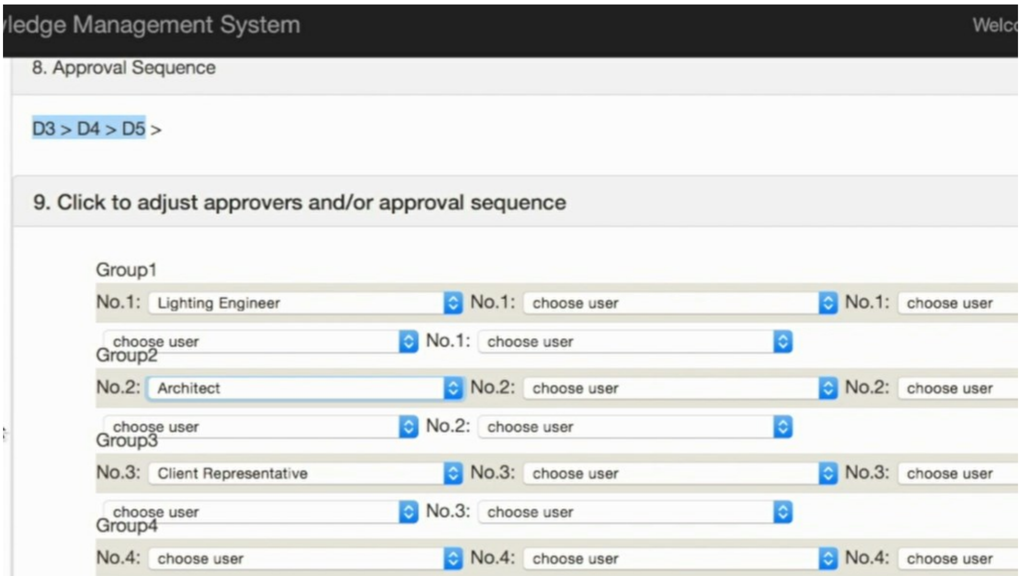

FIG. 6: User interface example – Adjust approval sequence.

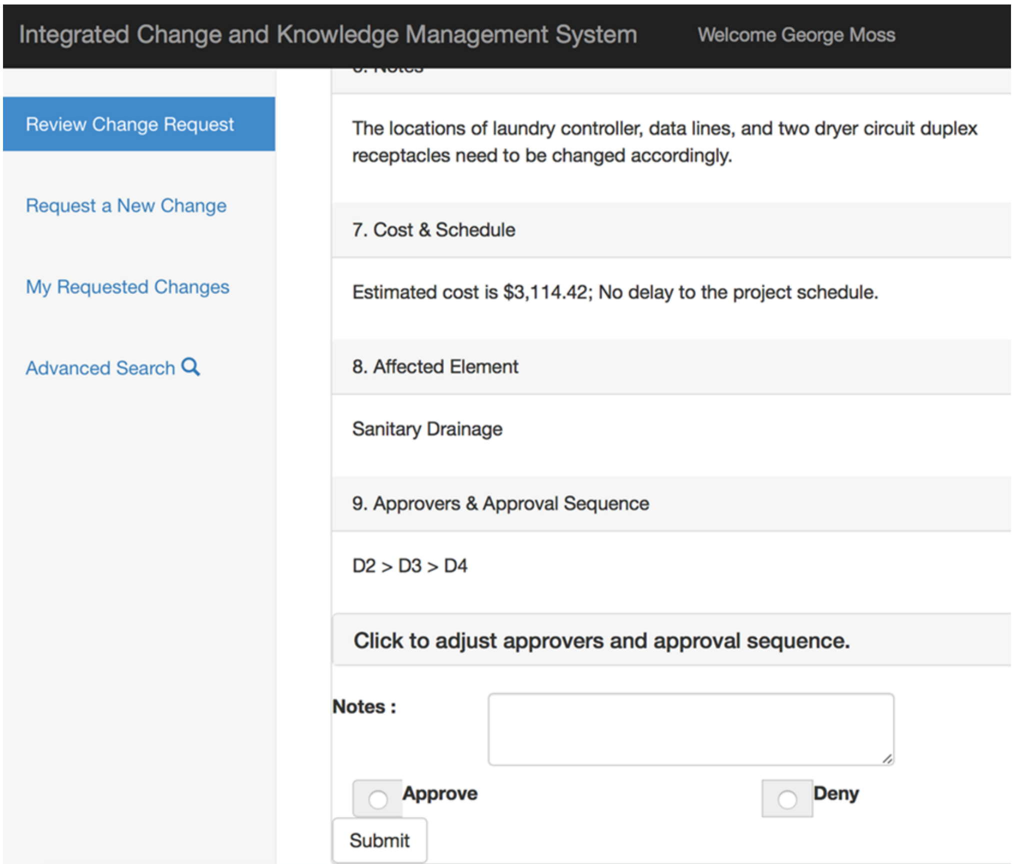

FIG. 7: User interface example – Change Approval page.

4. SYSTEM EVALUATION

Detailed evaluation was conducted to evaluate the performance and potential benefits of the ICKMS prototype. As the ICKMS is targeted to be used by all disciplines in a project team, thirteen industry practitioners who have experience of managing changes and/or project knowledge were selected. Table. 4 shows the role and experience of each evaluator.

In terms of the evaluators' experience of using information management systems in construction projects, 38.5% (5 evaluators) were very experienced and another 38.5% (5 evaluators) had some experience. 15.4% (2 evaluators) considered themselves novices/still learning, and the rest 7.7% (1 evaluator) had no experience at all.

TABLE. 4: Evaluator Roles and Experiences.

No.	Role	Experience of using information management systems	Workshop
1	Construction procurement	Change management systems	Workshop 1
2	Project manager	Change management systems	
3	Construction project manager	Change management systems& Knowledge management systems	Workshop 2
4	Facilities project manager	Change management systems& Knowledge management systems	
5	Architect	Change management systems	
6	Architect	N/A	
7	Architect	Knowledge management systems	
8	Arch designer	Knowledge management systems	
9	Assistant director	Change management systems& Knowledge management systems	
10	New construction commissioning	N/A	
11	Manager of design services	Knowledge management systems	
12	Construction contractor	Change management systems	
13	Designer code consultant	Change management systems	

4.1 Evaluation Objectives

The main objectives of the evaluations were to:

- Obtain user feedback about system's interface and ease of use, such as system navigation, learning and ease of access;
- Evaluate system functionality against the desired features;
- Evaluate the potential benefits of implementing the ICKMS in terms of management of changes and knowledge, decision-making, and energy retrofit design;
- Identify barriers and challenges of implementing the ICKMS in real construction projects; and
- Obtain suggestions from industrial practitioners for further improving the prototype system.

4.2 Evaluation Steps

The evaluation process is shown in Fig. 8. Evaluators were first given a brief introduction that included research objectives, the ICKM approach, and features and design of the ICKMS prototype. This step provided a general overview of the research and helped the evaluators understand the contents and objectives of the evaluation process. Following the introduction, the evaluators were asked to provide their basic background by completing a questionnaire. Next, a walkthrough of the ICKMS was presented with the prototype being hosted on a virtual website server. The system's configuration, functionality and user interface were explained and demonstrated to the evaluators. This step helped evaluators understand the process involved in interacting with the system from the viewpoint of end-users. The system walkthrough was followed by a demonstration of a change and knowledge management example that showed the evaluators a complete process to manage changes, dependencies and knowledge in a construction project. After the demonstration, a focus group discussion was conducted to discuss the system usability, benefits, and implementation barriers, as well as answering evaluators' questions. In the end, the evaluators were asked to complete a questionnaire to collect their feedback and opinions about the prototype's performance and effectiveness.

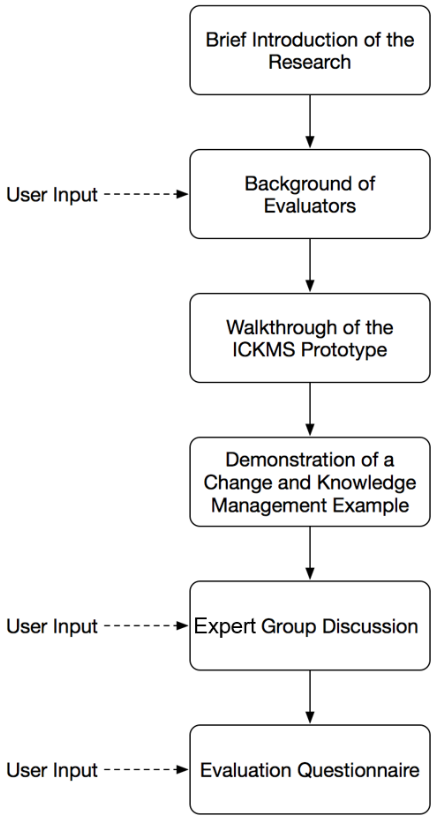

FIG. 8: Evaluation Steps.

4.3 Questionnaire Design

A questionnaire of 18 questions was developed to collect feedback from the evaluators. Table. 5 shows two samples questions. Both open-ended questions (to collect subjective feedback and background information) and closed questions (to collect evaluation ratings) were used in the questionnaire. To reduce the potential for bias, the following requirements were followed when developing the questionnaire (Taylor-Powell, 2008):

- Keep the order of choices the same in the questionnaire;
- Avoid using just numbers for the point scale;
- Balance the scales with an equal number of negative and positive choices; and
- Match response options to questions.

The questionnaire was divided into three parts:

- Evaluator profile: Collect evaluators' background information and experiences of using information management systems in construction projects;
- ICKMS evaluation: Collect feedback and ratings from evaluators in terms of user interface, system features, implementation barriers and potential benefits; and
- Overall evaluation and suggestions: Collect positive aspects and improvement suggestions.

TABLE. 5: Sample Question.

Questions	Choices (if any)	Category
How does the ICKMS handle dependency checking?	a) Very good b) Good c) Fair d) Poor e) Very poor	System Features
The ICKMS will help prevent loss of knowledge (especially change related knowledge) in a project.	a) Strongly agree b) Agree c) Disagree d) Strongly disagree	Potential Benefits

4.4 Evaluation Results

4.4.1 Ease of Use

Evaluators were asked to evaluate the prototype's ease of use in terms of navigating, learning and accessing the system, and provide their opinions on whether they 'strongly agree', 'agree', 'disagree' of 'strongly disagree' with the statements. Below are the collected responses:

i. It is easy to navigate the system (Agree: 100%);
ii. In my view, the ICKMS is easy enough for an average team member to learn and use in his/her daily tasks (Agree: 100%); and
iii. I think the Web-based environment makes ICKMS easy to access (Strongly Agree: 38.5%, Agree: 61.5%).

Overall, all the evaluators agreed (or strongly agreed) that the ICKMS prototype is easy to navigate, learn and access, and the system is user-friendly. 45% of them indicated that the system is very user-friendly.

4.4.2 System Features

Evaluators were asked to evaluate the core features with regard to checking dependencies, capturing lessons learned, tracking change approval status, and tracing change histories. Based on a 5-point scale ('very good', 'good', 'fair', 'poor', and 'very poor'), the responses are as follows:

i. How does the ICKMS handle dependency checking? (Good: 58.3%; Fair: 41.7%);
ii. How does the ICKMS handle lessons learned (knowledge) capture? (Very Good: 30.8%, Good: 38.5, Fair: 30.8%); and
iii. How does the ICKMS perform in tracking change approval status and tracing change histories? (Very Good: 38.5%, Good: 46.2%, Fair: 15.4%).

More than half of the evaluators (58.3%) agreed that the ICKMS is good at checking dependencies. Some evaluators were concerned about the linear process of change approval processes and suggested considering possible collaborations between different disciplines. It was suggested that the project manager be responsible for tracking change status and coordinating conflicts between disciplines. 69.3% of the evaluators agreed that the ICKMS is good (or very good) at capturing lessons learned. It was agreed by all evaluators that capturing lessons learned is one of the most important and innovative features of the prototype. It is also important to establish the process to evaluate and filter lessons learned, and fit the KM process into organizations. As for tracking change approval status and tracing change histories, 84.7% of the evaluators indicated that the ICKMS is good (or very good) at managing change information. Regarding the notification feature based on a 5-point scale ('extremely useful', 'very useful', 'moderately useful', 'slightly useful', and 'not at all useful'), all evaluators recognized its benefits, with 61.6% indicating that this feature is very or extremely useful.

4.4.3 Potential Benefits

Evaluators were asked to evaluate the potential benefits of ICKMS and provide their opinions on whether they 'strongly agree', 'agree', 'disagree' of 'strongly disagree' with the following statements:

i. In my opinion, the ICKMS will help the decision-making process in a project (Agree: 92.3%, Disagree: 7.7%);
ii. The ICKMS will help prevent loss of knowledge (especially change related knowledge) in a project (Strongly Agree: 23.1%, Agree: 76.9%);
iii. The implementation of such a system will help meet energy efficiency goals (Strongly Agree: 7.7%, Agree: 76.9%, Disagree: 15.4%); and
iv. I believe the implementation of the ICKMS will offer tangible benefits to the AEC Industry (e.g. reduction of workload, time and cost) (Agree: 100%).

The evaluators provided very positive feedback regarding potential benefits of the ICKMS prototype. It was agreed by most evaluators that the implementation of the ICKMS will prevent loss of knowledge and facilitate the decision-making process.

4.4.4 Implementation Barriers

92% of the evaluators agreed that there are challenges to implementing the ICKMS in energy retrofits. The following potential barriers were collected:

- The adoption of the ICKMS will add a new system to a project, which means that each team member needs to work with a separate user account. It is a common challenge for a project team when they have to deal with a new tool, and the evaluators suggested that the ICKMS can be further developed to integrate with existing project information management systems;
- There will be extra cost and efforts to implement the ICKMS. Organizations probably have been used to using existing software and may therefore be reluctant to invest money and associated training efforts in a new tool. It is necessary to plan the ICKM implementation from the very beginning of a project and estimate the required resources (financial responsibility, hardware/software support, and ownership of the system, etc.) for a successful adoption; and
- The need for person-to-person interactions should be accommodated along with the ICKMS implementation. It is risky to solely rely on information technologies, and it is critical to involve human factors in the system adoption process.

4.5 Discussion

The system evaluation received very positive feedback from industry practitioners. Some of the illustrative comments are:

- *"Love the ability to tie lessons learned to changes" (assistant director);*
- *"Great resource at the beginning of a job" (architect);* and
- *"Tracks the status and evolution of changes well" (manager of design services).*

The evaluators were asked the overall significance of the ICKMS in energy retrofit projects as well as the AEC Industry. All of them thought that the ICKMS will be a useful tool, and 70% of them agreed that the ICKMS will be very useful. During the evaluation process, the evaluators were asked if they were able to understand the research concepts and prototype functionality, and follow all the evaluation steps. Only 1 evaluator (7.7%) expressed the concern that it was not easy to understand the whole picture because he felt that limited information was presented. It was also suggested by several evaluators that the ICKMS can be tested in a real project for better evaluation results.

The following positive comments were highlighted by the evaluators:

- The ICKMS can capture lessons learned during CDM processes and prevent loss of knowledge. This feature provides a solution to formally document project knowledge. It also forces team members to keep lessons learned in mind so that knowledge can be collected and disseminated among team members;
- The Web-based environment provides easy access to project teams, and the intuitive interface can help users navigate the system efficiently;
- Tracking where a change request is at the approval stage is highly desirable. It can help project managers stay updated with each change request and hold people accountable in the decision-making chain; and
- The system provides a central database to track changes, document histories and store project knowledge. Team members can access the same repository throughout the construction workflows.

The evaluators showed great interest and expressed the desire to see more features in the system. The following suggestions were collected:

- Further enhance the ICKMS to interact with existing project information management systems. It is suggested to build an integration platform to share data between the ICKMS and other systems to avoid information duplication;
- The notification reminders (emails) should be more project-specific in the subjects and contents, because the system may be used by a project manager with different projects. For example, the project name should be added into the message contents;
- More information can be collected from change requests, such as deadlines and additional comments;
- Continue working on the graphic design to make the system more user-friendly. More graphic icons can be added in addition to word descriptions; and
- A mobile app is desirable to make the ICKMS accessible on mobile platforms. In some circumstances (such as approving/denying a request), a mobile app is preferable than a computer system because less redundant information will be displayed on a mobile device.

5. 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 capture change-related lessons learned or store both CDM and KM information in a structured data format. An Integrated Change and Knowledge Management System (ICKMS), which demonstrates the implementation of the ICKM approach from the perspective of IT applications, has been developed and presented in this paper. It helps integrate the CDM and KM processes, and provides a single platform for change and knowledge management.

The ICKMS is composed of two main components: (i) an automated change and dependency management system (CDMS) to implement the CDM process, and (ii) a knowledge management system (Capri.net) to implement the KM process. By integrating the CDMS and Capri.net, the ICKMS is able to track change request status, trace change history, manage lessons learned from change request, conduct dependency checking, and send out automatic notifications to team members. In addition, the Web-based feature makes ICKMS compatible with all types of operating systems and easy to deploy/maintain. As for limitations, although the prototype has been evaluated by industry practitioners, it has not been implemented in real projects. In future research, features such as automatic report generation, escalation based on priority and category, system administration center, and due day reminders can be incorporated into the system. It would also be helpful to improve the integrated system as a single platform that manages data in a central repository. Mobile applications can also be developed for portable platforms.

The main contribution of this research is demonstrating an integrated process from the system point of view to collect change and dependency information, and manage knowledge at the same time to guide future CDM activities. The ICKMS prototype provides an effective solution to manage changes, dependencies and knowledge throughout the lifecycle of a project.

ACKNOWLEDGEMENTS

This work was funded by the Department of Energy as part of the Energy Efficient Buildings Hub (EEB Hub) project. This paper is an extended version of work published in the 32rd international CIB W78 conference (Liu et al, 2015). We extend our previous work by presenting end user requirements, system design and development, system implementation and evaluation.

REFERENCES

Ackoff R. L. (1989). From data to wisdom, *Journal of Applied Systems Analysis*, 16, 3-9.

Anumba C. J. and Pulsifer D. (2010). Knowledge management systems for construction, *Proceedings of Construction Research Congress 2010*, ASCE, 687–696.

Anumba C. J., Egbu C. and Carrillo P. (2005). Knowledge management in construction, John Wiley and Sons.

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

Burnson F. (2015). Compare construction project management software, Retrieved in June 2015 from <<http://www.softwareadvice.com/construction/project-management-software-comparison/>>

CCPS. (2008). Guidelines for the management of change for process safety, *Center for Chemical Process Safety*, Wiley-Interscience, New York.

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

Davenport T. and Prusak L. (1998). Working knowledge: How organizations manage what they know, *Harvard Business Press*.

Elgobbi E. M. (2010). Exploring the perceptions of knowledge management and knowledge management systems: A case study on the Libyan oil industry, *Proceedings of 10 International Conference & Workshop on Emerging Trends in Technology*, Mumbai, Maharashtra, India, 2010, IEEE, 445-452.

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

Ibbs C., Wong C. and Kwak Y. (2001). Project change management system, *Journal of Management in Engineering*, 17(3), 159-165.

Jallow A. K., Demian P., Ruikar K., Daniels K., Machin C. H. and Palmer, C. (2011). Lost in transition: transition from design to construction: losses in knowledge and tools for their minimisation. Loughborough: School of Civil and Building Engineering, Loughborough University.

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

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

Kok G., Jongedijk S. and Troost, J. (2003). Insights from KPMG's European knowledge management survey 2002/2003, *KPMG White Paper*.

Lee S. and Peña-Mora F. (2007). Understanding and managing iterative error and change cycles in construction, *System Dynamics Review*, 23(1), 35–60.

Lin Y. C. (2014). Construction 3D BIM-based knowledge management system: a case study, *Journal of Civil Engineering and Management*, 20(2), 186-200.

Liu F. (2015). *Integrated change and knowledge management for energy efficient retrofit projects*, Ph.D. Thesis. December 2015, The Pennsylvania State University.

Liu F., Anumba C. J. and Jallow, A. K. (2015). An automated integrated change and knowledge management system, *Proceedings of 32nd CIB W78 Conference 2015, Eindhoven*, The Netherlands.

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

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

Liu Y., Ma L. and Huang S. (2009). Construct fault diagnosis model based on fault dependency relationship matrix, *Proceedings of 2009 Pacific-Asia Conference on Circuits, Communications and Systems*, PACCS 2009, IEEE, Washington, DC.

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

McCarthy T. J., Kahn H. J., Elhag T. M. S., Williams A. R., Milburn R. and Patel, M. B. (2000). Knowledge management in the designer/constructor interface, *Proceedings of 8th International Conference on Computing in Civil and Building Engineering*, 836-843.

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

Nonaka I., Toyama R. and Konno, N. (2000). SECI, Ba and leadership: a unified model of dynamic knowledge creation, *Long Range Planning*, 33, s5–34.

OmniClassTM. (2015). Retrieved in January 2015 from <<http://www.omniclass.org/>>

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

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

Stewart T. and Ruckdeschel C. (1998). Intellectual capital: the new wealth of organizations, *Performance Improvement*, 37, 56–59.

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

Taylor-Powell E. (2008). Working for rating scales, Retrieved in September 2015 from<< http://www.uwex.edu/ces/4h/evaluation/documents/Wordingforratingscales.pdf>>

Tuomi I. (1999). Data is more than knowledge: implications of the reversed knowledge hierarchy for knowledge management and organizational memory, *Proceedings of 32nd Annual Hawaii International Conference on Systems Sciences*, 1999, 70-80.