Creating a Knowledge Management Wiki for a Structural Engineering Firm

Erica Dekker
Master's Project Narrative
Lewis Department of Humanities
Illinois Institute of Technology
April 10, 2012

Table of Contents

Introduction 1

Knowledge Management in the Architecture, Engineering, and Construction Industry 2

Knowledge Management Needs of Robert L. Miller Associates 5

Using Wikis for Knowledge Management 7

Project Management 10

Feasibility Study 11

Usability Assessment 19

Content Management 24

Future Work 31

References 33

Appendix 1: Project Charter 36

Appendix 2: Revised Schedule and Deliverables 38

Appendix 3: Fall Semester Activity Log 40

Appendix 4: Wiki Table of Contents (Proposed) 44

Introduction

For my master's project, I project-managed the creation of a knowledge management wiki for a structural engineering firm. In the process, I researched the knowledge management (KM) needs of firms in the architecture, engineering, and construction (AEC) industry. After identifying the specific KM needs of Robert L. Miller Associates (RLMA), a structural engineering firm located in Rolling Meadows, Ill., I determined that a wiki would be the best system to address RLMA's needs. From the start of the project, I used established practices in project management, including performing a feasibility study to choose the best wiki engine for this use. I then assessed the usability of the chosen system (Confluence), created a template for the wiki content, and identified opportunities for future work.

Knowledge Management in the Architecture, Engineering, and Construction Industry

The information needs of the AEC industry are complex. Each project requires numerous information inputs, such as building codes and soil reports, and generates several outputs, such as architectural drawings, structural drawings, and shop drawings. Many firms in this industry have discovered that KM systems help their staffs cope with these information demands and provide a number of additional benefits.

Knowledge management systems can help construction professionals apply knowledge gained from one project to be applied to another.[1] Tserng and Lin explain that this reuse "minimizes the need to refer explicitly to past projects, reduces the time and cost of solving problems and improves the quality of solutions during the construction phase of a construction project."[2] They also note that improving problem solving decreases the likelihood of repeat problems. In addition, Udeaja et. al. suggest that this knowledge gleaned from past projects can be used for training new employees.[3]

Even more powerful than a KM system that facilitates knowledge reuse is one that can tap into tacit knowledge.[4] Egbu emphasizes the importance of tacit knowledge, noting that "competitive advantage stems from the firm-specific configuration of its intangible knowledge."[5] Woo et. al. suggest that much knowledge in the AEC industry is tacit because

---

[1] Dave and Koskela, "Collaborative Knowledge Management."
[2] Ping Tserng and Lin, "Developing an Activity-based KM System."
[3] Udeaja et al., "A Web-based Prototype."
[4] Dave and Koskela, "Collaborative Knowledge Management."
[5] Egbu, "Managing Knowledge and Intellectual Capital."

the work is project based.[6] They describe, "Throughout the whole life cycle of a construction project, AEC firms rely on their experiences, professional intuition, and/or other forms of tacit knowledge to accomplish satisfactory work."[7] Tserng and Lin explain the danger of leaving this knowledge untapped: "when the engineers and experts complete projects or leave the company, they normally take domain knowledge with them and leave little or nothing that will benefit subsequent projects or the company."[8]

Capturing tacit knowledge is not easy, but a growing number of KM systems are able to do so. Dave and Koskela note that social networking applications like forums and wikis are increasingly being used to make tacit knowledge explicit.[9] Tserng and Lin recommend activity-based KM systems for capturing tacit knowledge.[10]

When KM systems are successful in capturing tacit knowledge and facilitating knowledge reuse, competitive advantages emerge.[11] For example, "a general contractor may make innovative use of knowledge generated and accumulated through project activities, and share it across related projects to be competitive."[12] Udeaja et. al. state that "collaboration through learning and web-based applications will provide the competitive edge that enables all the participants in a project development to prevail and grow."[13] The result of these competitive advantages should be increased profits for the firm, which

---

[6] Woo et al., "Dynamic Knowledge Map."
[7] Ibid.
[8] Ping Tserng and Lin, "Developing an Activity-based KM System."
[9] Dave and Koskela, "Collaborative Knowledge Management."
[10] Ping Tserng and Lin, "Developing an Activity-based KM System."
[11] Woo et al., "Dynamic Knowledge Map."
[12] Ping Tserng and Lin, "Developing an Activity-based KM System."
[13] Udeaja et al., "A Web-based Prototype."

corresponds to Egbu's definition of KM: "how knowledge is effectively managed to produce profit in an organisation."[14]

The potential of KM for achieving competitive advantage is tied to KM's power to stimulate innovation. When learnings from previous projects are captured in a KM system, reflection on that learning can foster innovation.[15] Egbu lists "the ability to manage organizational knowledge (tacit and explicit) and build knowledge enhancing approaches, systems and technology" among the critical success factors for innovation.[16] More than half of respondents in a study by Egbu indicated that KM has contributed to a "new technology that has internal benefits to the company."[17]

However, for innovation to occur, the KM system must be used and the knowledge it contains must be kept current.[18] Egbu warns that "KM that focuses on creating network structures to transfer only explicit knowledge will be severely limited in terms of its contribution to innovation and project success."[19]

---

[14] Egbu, "Managing Knowledge and Intellectual Capital."
[15] Udeaja et al., "A Web-based Prototype."
[16] Egbu, "Managing Knowledge and Intellectual Capital."
[17] Ibid.
[18] Dave and Koskela, "Collaborative Knowledge Management."
[19] Egbu, "Managing Knowledge and Intellectual Capital."

Knowledge Management Needs of Robert L. Miller Associates

Robert L. "Bob" Miller founded the structural engineering firm Robert L. Miller Associates in 1984. The firm is located in Rolling Meadows, Ill. Brian Dekker joined the firm in 2009 and is in the process of purchasing the firm from Bob, with full ownership to be transferred in late 2012 or early 2013.

From 2009 to 2011, the firm grew from two to six engineers. The new engineers had varying amounts of previous project experience. A few quickly acclimated and could work independently, but others needed more hands-on guidance from Bob and Brian. Occasionally, work is duplicated when the engineers don't realize that a standard already exists for a particular situation. In addition, Brian expressed concern that the growing staff's differing styles would lead to inconsistencies in their work product and client experience. Some engineers file documents one way, and others have their own idiosyncratic method. This creates difficulties when projects are handed from one engineer to another.

At the same time the staff was growing, Bob was reducing his hours. He will retire from day-to-day involvement in the firm by 2013. Before that happens, Brian wants to capture Bob's wisdom and best practices, realizing the wealth of tacit knowledge that Bob has gained over the years.

Brian decided that an office standards manual should be created to address these needs. This manual would document the procedures involved from beginning to end of a structural project, including communicating with clients, making calculations, completing drawings, and archiving files. It would include links to or screenshots from the applicable

building codes and standards, so that the engineers would not have to waste time refinding this information every time they need it.

Brian contacted me to request that I oversee the manual's production. Together we identified the following requirements:

1. The manual must be digital, enabling searching by keyword.
2. The manual must be organized by topic with cross-linking between topics as appropriate.
3. The manual must be easily editable without requiring specialized knowledge or software.
4. The manual must be protected from unauthorized access.

With those requirements in mind, I considered several possible formats. One option was to create the manual in Adobe InDesign and convert it to a PDF, so that it could be easily searched and bookmarks could be used to cross-link between topics. This option would not, however, meet the requirement of being easily editable. Another option was to hand-code an internal website in HTML and house it on the company server. This would protect the manual from unauthorized access, and it would be digital and cross-linked, but again, editing would be difficult for Brian and the other engineers who do not know HTML. The format I eventually chose was a wiki.

Using Wikis for Knowledge Management

Our specific requirements made a wiki a logical choice. A wiki could be installed on the firm's server, so that it could not be accessed by those outside the firm. Because it would run in a browser, no specialized software would be required for users to access and edit it. The basics of navigating and editing the wiki would already be familiar to the staff from previous Internet use. In addition, the hyperlinking and search capabilities would enable the mix of searchability and organization that we desired.

In addition to meeting our basic requirements, wikis have powerful features that can be leveraged for effective knowledge management. A wiki allows the information to be arranged hierarchically and navigated like a network. The tension between information hierarchies and networks is discussed in Alex Wright's book *Glut*.[20] He notes that "wherever networked systems take root, they disrupt the old hierarchical systems that preceded them," but he emphasizes that the common view of hierarchical systems as "restrictive, oppressive tools of control" is too simplistic.[21] Instead, networks and hierarchies "not only coexist, but they are continually giving rise to each other"[22] and are in fact complementary. Wright quotes John Seely Brown and Paul Duguid, who state that self-organization and formal organization "perform an intricate (and dynamic) balancing act, each compensating for the other's failings.

---

[20] Wright, *Glut*.
[21] Ibid., 6–7.
[22] Ibid., 8.

Self-organization overcomes formal organizing's rigidity. Formal organization keeps at bay self-organization's tendency to self-destruct."[23]

This balance is reflected in the wiki system's editing capabilities. While Brian and Bob are supplying the initial framework and content for the wiki, any authorized engineer will later be able to edit it. Should mistakes be made, they can be easily corrected by rolling back to a previous version. A wiki gives the leaders a measure of control without stifling peer-to-peer contributions.

In *Infotopia*, Cass R. Sunstein describes additional benefits of wikis for knowledge management. In discussing Wikipedia, he notes that "it isn't easy to write an entry from scratch, especially on a technical topic. But if thousands of people are in a position to make small additions and improvements, an initial skeleton can rapidly become a full body."[24] RLMA does not have thousands of engineers to contribute to the wiki, but the staff's contributions will help flesh out the initial content. Through its features that make it easy to add and edit content, a wiki "provides an exceptional opportunity to aggregate the information held by many minds."[25]

Sunstein also notes a number of conditions in which wikis are not successful: "if vandals are numerous, if contributors are confused or prone to error, or if people are simply unwilling to devote their labor for free."[26] In this context, in a workplace, free from anonymity, vandalism seems unlikely. In contrast, contributor error is more of a concern, which is why

---

[23] Brown and Duguid, *The Social Life of Information*, 170.
[24] Sunstein, *Infotopia*, 153.
[25] Ibid., 195.
[26] Ibid.

Brian and Bob are closely overseeing the content. A numeric typo or mistake in calculation could have costly or even dangerous consequences, so edits made in the system will be checked to ensure that errors have not been introduced.

The final condition Sunstein lists is also a concern: how to motivate the staff to contribute. It is our hope that the system will prove sufficiently useful to all staff members that they have a motivation to keep it up-to-date and make it comprehensive. It's possible, however, that this motivation will not be enough. Lutters, Ackerman, and Zhou note a study of the adoption of Lotus Notes in a consulting firm: the firm managers wanted everyone to share their expertise using the system, but the consultants, motivated by the need for billable hours, had little reason to do so.[27] If reward systems are realigned, however, these conflicts need not occur.[28] It's also necessary to establish trust in the wiki as the place to store information: "users need to trust that the system will be there in the future, or they will store the information in their personal space instead."[29] I will discuss these challenges further when I describe future work.

---

[27] Lutters, Ackerman, and Zhou, "Group Information Management," 239–240.
[28] Ibid., 240.
[29] Ibid., 244.

Project Management

Once Brian and I agreed on the requirements and wiki format, I began a formal project management process. I first wrote a project charter (Appendix 1) to clarify the goal, parameters, and schedule of the project. I adapted a format recommended by the Project Management Institute.[30] The schedule outlined in the charter was not very specific, because at the time we had only a general concept of the wiki contents. After the table of contents was determined, I created a revised schedule that broke down the milestones for Brian and me (see Appendix 2).

I kept an activity log to keep a record of our discussions and progress. This log (Appendix 3) covers the fall semester, August through December 2011, during which the wiki system was chosen, templates were developed, and initial content was uploaded. Since then, content development has slowed as RLMA experienced several staff turnovers, and Brian and Bob have not had time to write the content.

To select the wiki system, I conducted a formal feasibility study. Since the system we chose would have a significant impact on how easy the wiki was to use, and thus how likely it was to be adopted, I wanted to follow a careful process and evaluate many alternatives. Because I also used the feasibility study to fulfill a course requirement in COM 523: Communicating Science, I wrote a proposal for the study. For that proposal, I created a Gantt chart (Figure 1) showing my schedule and process for the study.

---

[30] Project Management Institute Staff, *PMBOK Guide*, 77.

Tasks	Time						
Identify wiki engines							
Finalize evaluation criteria							
Choose three to five options							
Assess options							
Write report outline							
Prepare draft/presentation							
Prepare final report							
	9/19-23, 2011	9/26-30	10/3-7	10/10-14	10/17-21	10/24-28	10/31-11/4

Figure 1. Gantt chart used in feasibility study proposal

Feasibility Study

Over the course of the feasibility study, I had three primary tasks: (1) identify the various wiki engines available to businesses, (2) establish criteria for evaluating the engines, and (3) assessing the engines using these criteria. I then conducted a cost-benefit analysis and formed my conclusions and recommendations.

Task 1: Identify the various wiki engines available to businesses

To identify the options available, I used the review site WikiMatrix,[31] which has a variety of tools to search and compare wiki engines. I decided to use two of the features: "Choice Wizard" (which asks a series of questions about your wiki needs and then generates a list of options) and the "25 Most Popular" feature comparison. I did both because it is possible for feature lists to become out of date, and I did not want a popular engine to be eliminated by Choice Wizard if the information was incorrect. In an attempt to focus on the most promising

[31] "WikiMatrix."

options, I decided to visit the websites of all of the 25 most popular engines, but I would only visit the websites of the engines selected by Choice Wizard if they appeared to meet the majority of the desirable criteria.

First, I used Choice Wizard to select engines that supported page history (necessary for error recovery) and could be installed (as opposed to externally hosted) and received 103 results. I looked through these results and eliminated those that did not support all the necessary criteria (listed in Table 1) or the most important desirable criterion: searching attachments.

Next, I visited the websites of the remaining results and eliminated those whose websites were not in English, those whose communities seemed no longer active, or those whose target audience seemed to be developers rather than business users. I also eliminated those that were also in the 25 Most Popular list, because I would be evaluating those separately. In the end, the Choice Wizard yielded three systems for more careful evaluation.

I then turned to the 25 Most Popular list. Of the 25, 2 were available only as hosted solutions (not self-hosted), so I eliminated those. I visited the websites of the rest of the most popular engines and eliminated those that did not seem targeted to business users and those that did not appear to meet at least most of our desirable criteria. At the end of this process, three engines remained for further evaluation.

In summary, the two tools provided six engines for me to evaluate formally.

Task 2: Establish criteria for evaluating the engines

In my feasibility study proposal, I had identified four necessary criteria and four desirable criteria. In the course of the study, I modified one criterion and added one additional criterion. The changes and addition are indicated in bold in Table 1. The modified criterion was originally that "the engine should be able to search image text (via optical character recognition or similar means)." I had to change that criterion because I was unable to find any engines that could do so. I was, however, able to find some that could search attached files such as PDFs and Microsoft Office documents. Therefore, I did not eliminate engines from further evaluation if they could not search images, but I did eliminate them if they could not search any type of attachment. The added criterion was motivated by a conversation with Brian, in which he emphasized the importance of easily being able to input content without concerning himself with formatting. Accordingly, I decided to add the criterion that the engine should support page templates.

Necessary Criteria	**Desirable Criteria**
• The engine must be available for self-hosting, as the information the wiki will contain is proprietary and should not be publicly accessible. • The engine must be compatible with the firm's server (Windows Small Business Server 2011). • The engine must enable attachments, such as PDFs. • The engine must, by default or through plug-ins, support input of mathematical formulas.	• The engine **ideally** should be able to search image text (via optical character recognition or similar means) **or at least be able to search attachments.** • The engine should facilitate exporting the content should a different system be used in the future. • The engine should be free or low-cost (less than $500). • The engine should be easy to install. • **The engine should support page templates.**

Table 1. Criteria used for evaluating wiki engines

Task 3: Assess the engines using these criteria

Once the criteria were identified, I assigned an importance value to each desirable criterion, ranging from highly important ("5") to least important ("1"). This evaluation method is similar to the "must have," "should have," "nice to have" system recommended by Ann Rockley.[32] Next, I assigned a score ranging from best ("5") to worst ("1") to each wiki engine option for each criterion. Finally, I multiplied the score for each criterion by its importance and calculate the total for each option. As seen in Table 2, Confluence[33] received the highest score.

Criteria	Importance	Confluence	XWiki	Share Point	Doku Wiki	Same Page	Screw Turn
Search image text or attachments	5	3	3	1	2	3	1
Export content	4	4	4	2	2	4	1
Free or low-cost	4	5	5	5	5	0	5
Easy to install	3	2	3	5	3	2	4
Create page templates	3	5	3	3	3	5	3
	Totals	72	69	57	56	52	50

Table 2. Evaluation of wiki engines

[32] Rockley, Kostur, and Manning, *Managing Enterprise Content*, 272.
[33] "Content and Social Collaboration Software."

Cost and Benefit Evaluation

As demonstrated in Table 2, the three wiki engines receiving the best numeric scores were Confluence, XWiki,[34] and Microsoft SharePoint.[35] In the following section, I discuss the ratings each of these three systems received against the criteria.

Confluence

Search image text or attachments. Because no systems were able to search image text as we had hoped, no system received more than a "3" for this criterion. Confluence received a "3" for its ability to search PDFs, Microsoft Office attachments, and image file names.[36,37]

Export content. Confluence received a "4" for this criterion, because while a direct import/export utility is not mentioned on the Confluence website, the content can be exported as XML, PDF, or HTML.[38]

Free or low-cost. Confluence received a "5" for cost because it is only $10 a year for up to 10 users.[39] A user is considered anyone who can log in to the system.[40] This cost includes support.

Easy to install. For ease of installation, Confluence received a "2" because a separate database installation is required. The Confluence website does, however, have detailed instructions,[41] and, as noted, support is available.

---

[34] "XWiki."
[35] "Create and Edit a Wiki."
[36] "Full Features List."
[37] "Searching Confluence."
[38] "Full Features List."
[39] "Confluence - Pricing."
[40] "Confluence Licensing."

Create page templates. Confluence received a "5" for this criterion, as it has a built-in template ability[42] as well as a plug-in that can turn the basic wiki page into a form to fill out.[43]

XWiki

Search image text or attachments. Like Confluence, XWiki received a "3" for searching image text because while it cannot search image text, it can search attachments such PDFs.[44]

Export content. XWiki has similar export abilities as Confluence, with one additional export format (RTF),[45] so it also received a "4."

Free or low-cost. XWiki is free and open source,[46] so it received a "5" for cost. Unlike Confluence, however, no professional support is included. While significant community support resources are available ,[47] professional support would cost a minimum of €3,000.[48]

Easy to install. One area where XWiki exceeded Confluence's abilities is in ease of installation, as no separate database is needed.[49] Therefore, XWiki received a "3" for this criterion.

Create page templates. XWiki received a "3" for this criterion, because while it is possible to create templates,[50] the instructions are lengthy and complicated.[51]

---

[41] "Installing Confluence on Windows."
[42] "Working with Templates."
[43] "Scroll Wiki Forms."
[44] "Search Application - XWiki."
[45] "Exports - XWiki."
[46] "License - XWiki.org."
[47] "Support - XWiki.org."
[48] "XWiki Support - XWiki.com."
[49] "Installation - XWiki."
[50] "Document Templates & Forms - XWiki."
[51] "Creating a FAQ Application - XWiki."

Microsoft SharePoint

Unlike the other systems I evaluated, Microsoft SharePoint is not primarily a wiki, but it is possible to create a wiki within the SharePoint system. Perhaps as a result of this, it was difficult to find information on SharePoint's wiki capabilities directly from Microsoft. This would have eliminated the system from further consideration if not for the fact that the system was already installed on the firm's server.

Search image text or attachments. Although I was not able to confirm this from Microsoft, SharePoint appears to lack the ability to search image text or attachments,[52] so it received a "1" on this criterion.

Export content. I encountered conflicting reports[53] regarding whether or not the content could be exported, so I gave it a "2" for this criterion.

Free or low-cost. As noted, the firm already has SharePoint, so it received a "5" for this criterion.

Easy to install. As noted, SharePoint is already installed on the firm's server, so SharePoint received a "5" for this criterion.

Create page templates. I gave SharePoint a "3" for page templates, because while the system allows master pages,[54] I was not able to determine how easy it would be to adopt multiple master pages within the wiki portion of SharePoint.

---

[52] "WikiMatrix / SharePoint Wiki Plus Features."
[53] Ibid.; "Export Your SharePoint Wiki"; "How to Export All Files from a Sharepoint Site?".
[54] "Introduction to SharePoint Master Pages."

Conclusions and Recommendations

Confluence met all the necessary and desirable criteria at very little cost. It also has a few unique features not quantified in the criteria, such as enabling images to be pasted into pages.[55] XWiki also did well against the criteria, but it lacked the guarantee of support that Confluence has. The primary advantages of the Microsoft SharePoint system were that it was already installed on the firm's server and thus free to use. Its features, however, were more limited, and the lack of clarity regarding support for page templates was especially concerning. For these reasons, I recommended the Confluence system. It received the highest score against the criteria, and Brian was impressed with its abilities. In particular, we anticipated that the ease of creating and using page templates would greatly diminish the time and difficulty involved in entering information into the system, making systematic adoption much more likely.

[55] "New Features in Confluence Wiki."

Usability Assessment

Outside of the scope of the feasibility study, I also assessed the usability of Confluence. Although I did not conduct any formal usability tests, I considered a variety of design principles. Kim Vicente notes that "the design of technology has to have affinity with human nature to be successful…. Design should begin by identifying a human or societal need—a problem worth solving—and then fulfill that need by tailoring the technology to the specific, relevant human factors."[56] In this scenario, there are a limited number of individuals who will be using the system, so the design must be tailored to them, not to an imagined average user. My goal was to see if Confluence would be usable for Brian and the other engineers, and thus far, it has been.

Rubin and Chisnell state that "to be usable, a product or service should be useful, efficient, effective, satisfying, learnable, and accessible."[57] Confluence has certainly proven useful. As I describe later, I was able to use it to create templates, and Brian was able to begin populating the templates with content. It is efficient and learnable, because we were quickly able to begin using it successfully. We have not experienced any technological bugs with the system, so it has behaved as expected, a measure of effectiveness.[58] It is also satisfying. Brian expressed pleasant surprise at its image-handling capabilities, and we both have positive feelings toward the system. Finally, we have encountered no barriers to accessibility thus far.

---

[56] Vicente, *The Human Factor*, 45.
[57] Rubin and Chisnell, *Handbook of Usability Testing*, 4.
[58] Ibid.

One reason for Confluence's usability is its transparency. Victor Kaptelinin and Bonnie A. Nardi describe transparent interaction as "an interaction in which the user can focus on his work, while the system—the mediating artifact—remains 'invisible.'"[59] They note that any system can become transparent "provided that sufficient time and effort is invested by the user,"[60] but given the particular users of this system—non-programmers who wanted a system that they could use without training—we needed a more readily transparent system. Confluence lends itself to transparency through its adherence to the design principle of mapping.[61] Lidwell, Holden, and Butler describe good mapping as when "the effect corresponds to expectation."[62] A good example of this principle in Confluence is when a user tries to insert a photo. All the user has to do is drag or paste the photo onto the page, and it is automatically uploaded and placed on the page. This movement is much more natural than having to navigate a menu to insert a picture, and it allows the user to seamlessly transition from text to image input and back again.

Confluence also embodies another design principle described by Lidwell, Holden, and Butler: control. This principle dictates that "the level of control provided by a system should be related to the proficiency and experience levels of the people using the system."[63] This is similar to the principle of progressive disclosure, which can be described via this example:

---

[59] Kaptelinin and Nardi, *Acting with Technology*, 79.
[60] Ibid.
[61] Lidwell, Holden, and Butler, *Universal Principles of Design*, 152.
[62] Ibid.
[63] Ibid., 64.

> "Infrequently used controls in software interfaces are often concealed in dialog boxes that are invoked by clicking a *More* button. People who do not need to use the controls never see them. For more advanced users, the options are readily available. In either case, the design is simplified by showing only the most frequently required controls by default, and making additional controls available on request."[64]

In Confluence, the user interface for creating and editing pages (see Figure 2) is what-you-see-is-what-you-get (WYSIWYG), which is easy for Brian and the other engineers to use. The user interface for making page templates is more complicated in that it uses wiki markup rather than WYSIWYG controls (see Figure 3), but since I am the primary user of that interface, this has not been a problem. The one shortcoming in this area is that there is no way to switch to wiki markup controls from the WYSIWYG on the standard page-edit screen. Ideally, the system would support this function for more advanced users.

As a whole, however, we have been satisfied with the usability of Confluence.

---

[64] Ibid., 188.

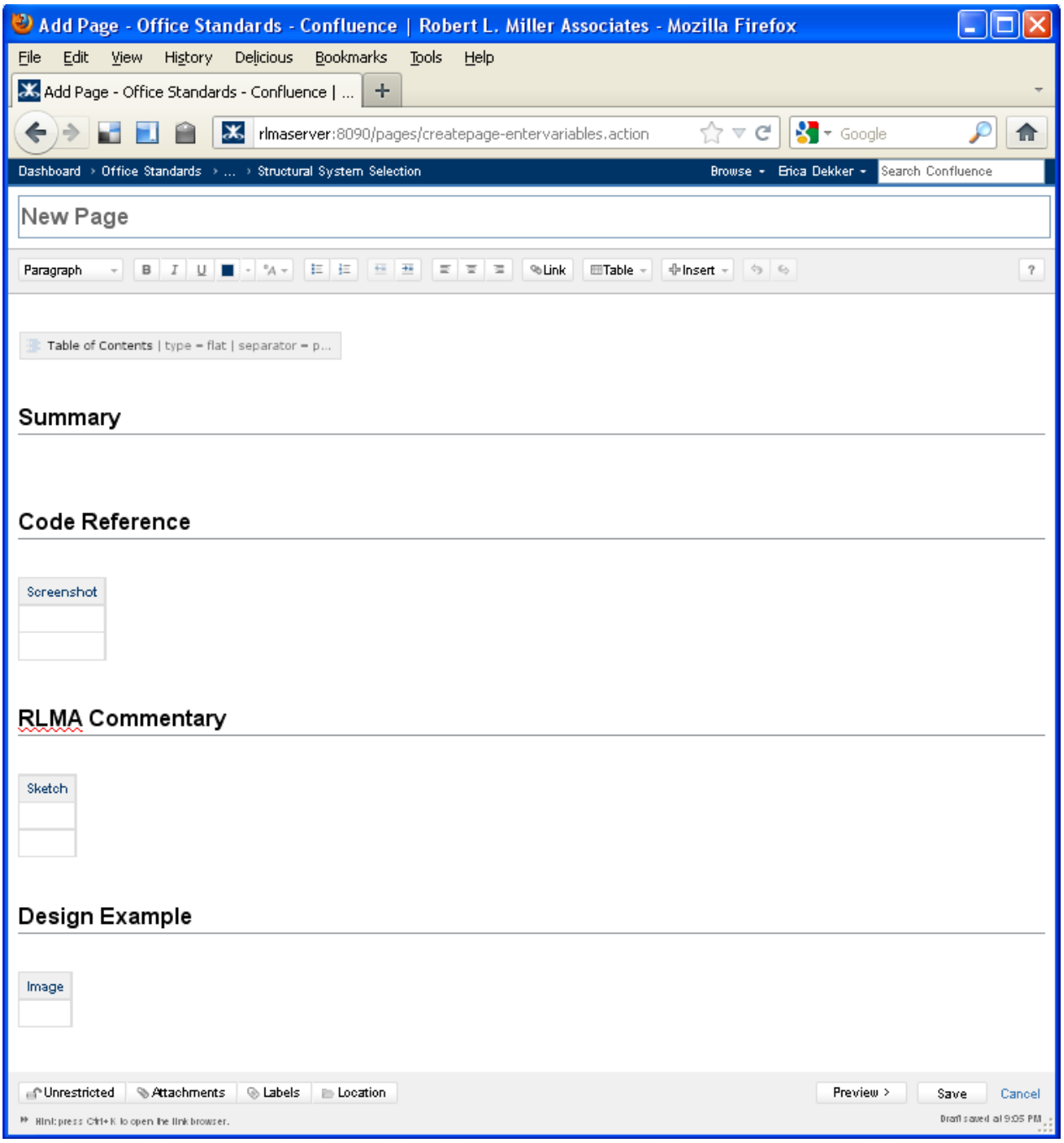


Figure 2. User interface for creating a new page from a template (WYSIWYG)

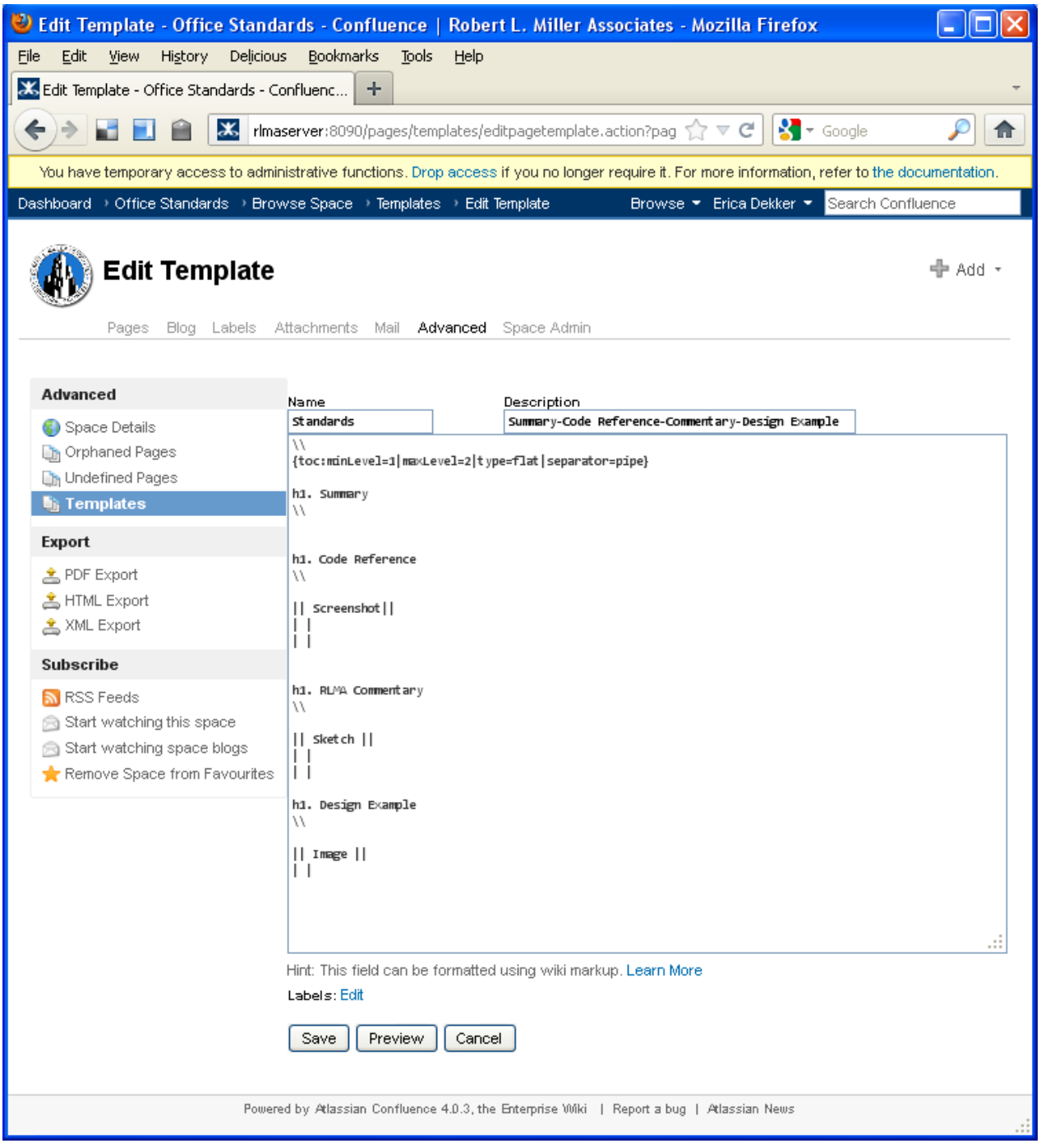


Figure 3. User interface for creating a template (uses wiki markup)

Content Management

While I was researching wiki engine options, Brian began developing the content that the wiki would house. He produced a table of contents, which can be found in Appendix 4. After we installed Confluence on the firm's server, I began creating the navigation based on the table of contents. The navigation is found on the left side of each page (see Figure 4), so it's easy for users to see related topics and skip from page to page. In addition to using the side navigation, users can also see a list of recently updated pages on the home screen when they log in. Yet another form of navigation is through the search feature, which is available on every page.

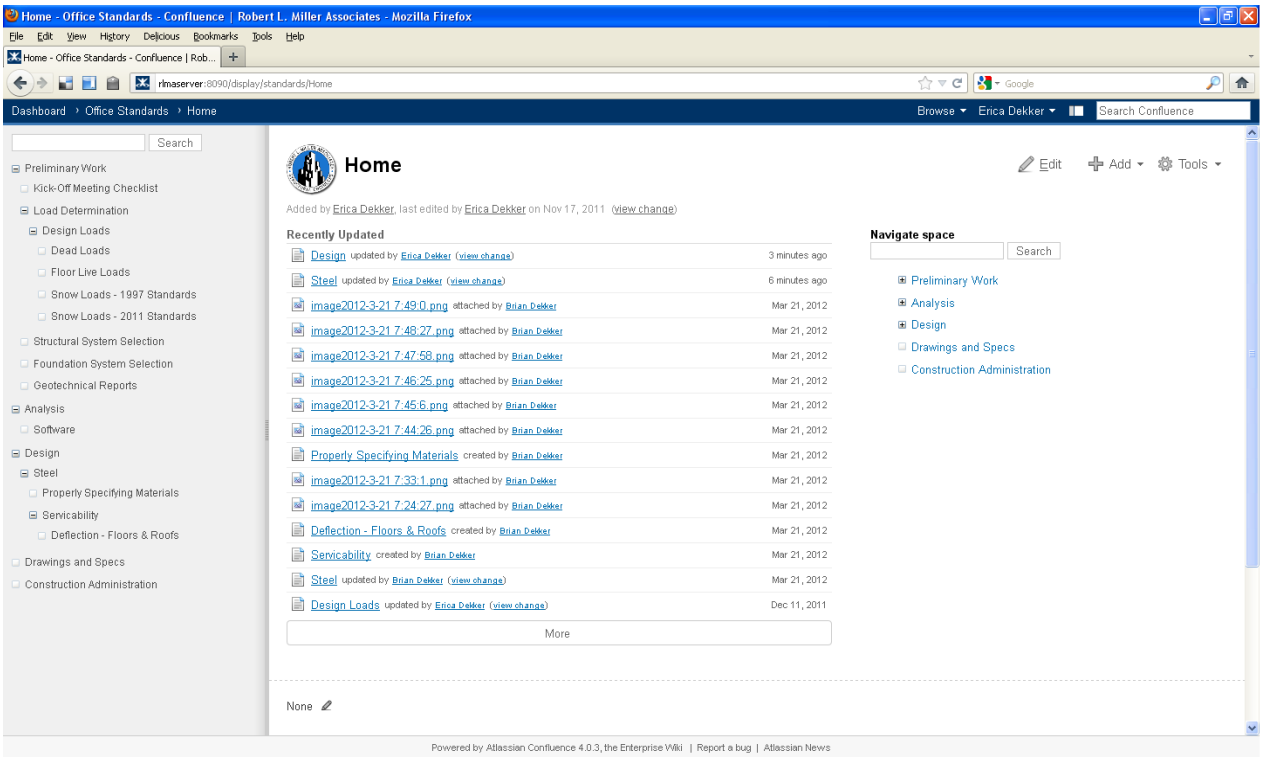


Figure 4. Home screen shown when logged into the wiki

The table of contents is organized by category, serving as a form of categorization metadata.[65] The search feature reduces the need to formally assign metadata to assist in retrieval,[66] but a tagging feature (called "Labels") exists should this be desired in the future. Rockley indicates a preference for using a controlled vocabulary for tags in order to ensure consistency,[67] and Confluence does have this capability.

In addition to the table of contents, Brian also provided a sketch of a template that he thought would fit the majority of wiki entries (see Figure 5). Using the sketch as inspiration, I created several template options in Confluence for Brian to review. We landed on the template shown in Figure 2. The template serves as an information product model, which specifies the type of information that should be included (as applicable) in each entry.[68] This establishes

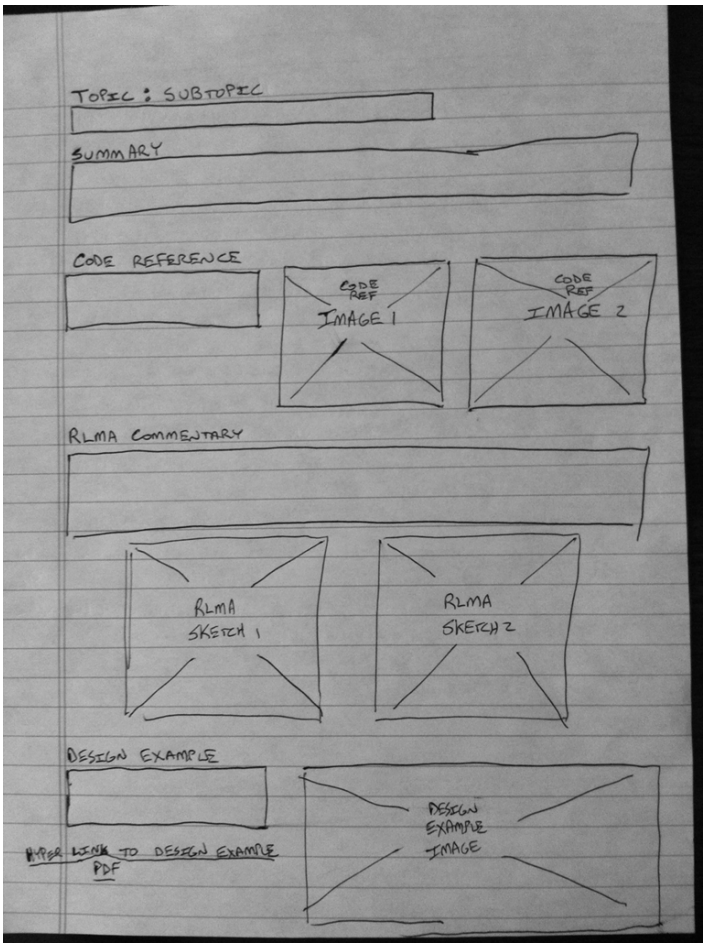


Figure 5. Sketch of desired template

[65] Rockley, Kostur, and Manning, *Managing Enterprise Content*, 186.
[66] Ibid., 195.
[67] Ibid., 199.
[68] Ibid., 168.

consistency from entry to entry and reduces the apprehension Brian and the other engineers might feel at facing a blank page. At the same time, any of the sections can be deleted and new sections added if a particular topic does not lend itself to the template.

This flexibility is important. Clay Spinuzzi describes the necessity of open systems: "Closed systems often inhibit the many intersecting activities in which workers must function. To get their jobs done, workers must adapt additional genres, reinterpret parts of the closed system in light of their own activities, and sometimes even abandon the closed system altogether."[69] In contrast, an open system acknowledges that users must be able to make alterations. Spinuzzi notes, "An open system can consist of an officially designed core that provides openings for workers' contributions. The point is not to rescue workers with a better designed system, but to provide a base for workers to build on."[70] This project has focused on creating the "officially designed core," but as I discuss in Future Work, I hope that the wiki becomes a "nexus for workers' innovations."[71]

To date, we have begun to populate content in the first section, "Preliminary Work," and the third section, "Design." In the original schedule, we had hoped to have starter content in all five sections by April 6, 2012, but this proved overly ambitious. As I mentioned previously, the turnover of several staff members has limited the time that Brian and Bob have to devote to this project, so they have not be able to deliver much content. The system is in place, however, when they are again able to contribute.

---

[69] Spinuzzi, *Tracing Genres Through Organizations*, 204.
[70] Ibid.
[71] Ibid., 205.

Of the initial content, a few entries fit almost exactly in the template as planned (see Figure 6), but most were adapted in some way.

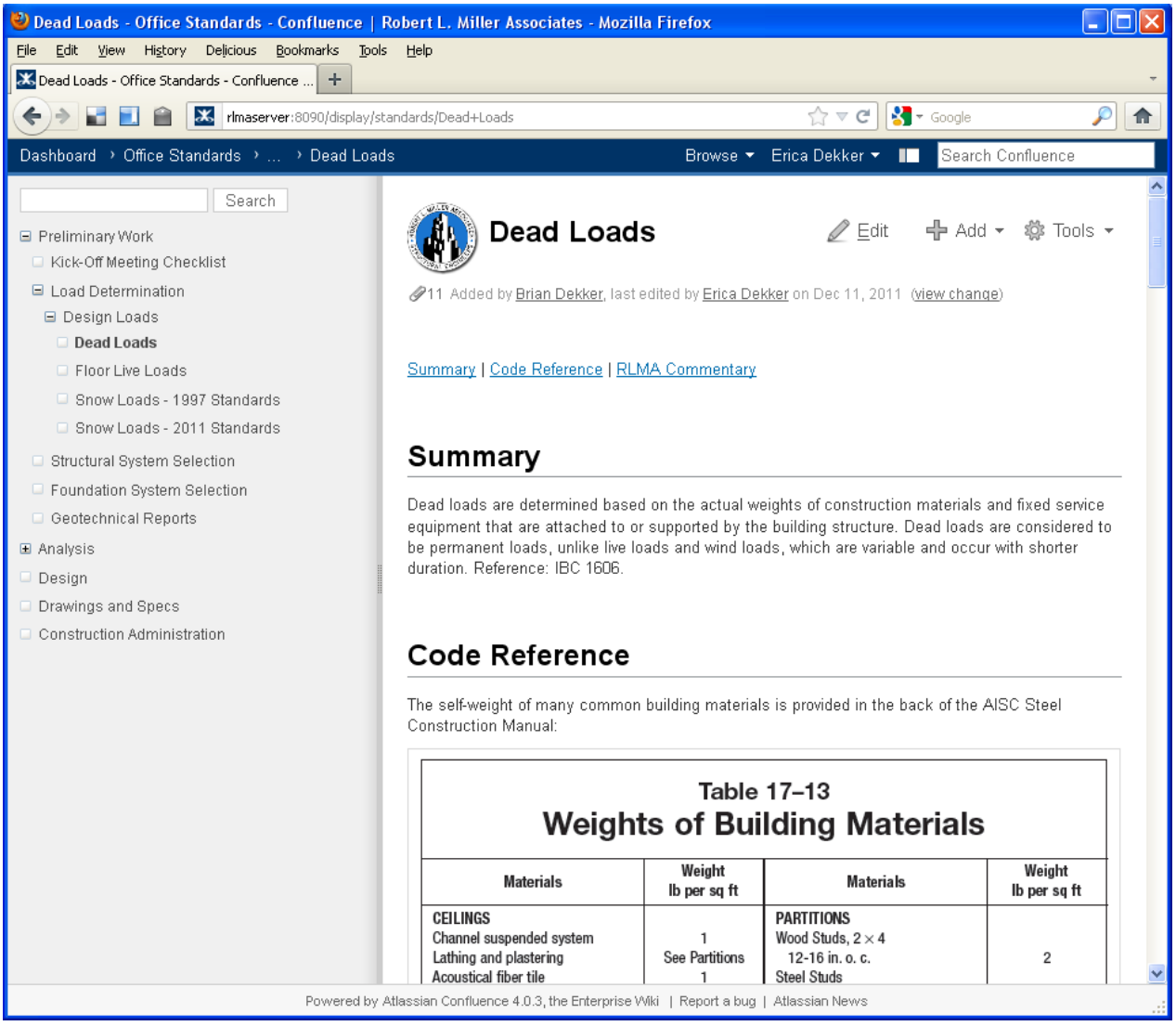


Figure 6. Sample entry adhering to the template

One item of content that fell outside of the template—in fact, outside of the format of a wiki entry—is the kick-off meeting checklist. This is a document that the engineers take to a

meeting with the architect at the start of each construction project. It lists the items of information that RLMA must have in order to begin work on the project.

Brian supplied the document that had been used in the past (see Figure 7). While the content was sufficient, the design of the document had clear opportunities for improvement. The words in all capital letters were difficult to read, given the lack of distinctive word shape[72] and excess of white space between the letters.[73] In addition, there was little to distinguish the pieces of information that needed to be checked off (e.g. to acknowledge receipt of an item) from those that needed to be filled in (e.g. the name of the general contractor).

I redesigned the document to increase its aesthetics and usability (see Figure 8). I chose to use Futura, a font that has five of the six features of a professional font, as identified by Riley and Mackiewicz,[74] resulting in a professional-looking document that is not overly formal. I replaced the all-caps with a mix of uppercase and lowercase letters to increase readability. I also removed the underlining from the column headings, because underlining is less effective than using bolded letters to create contrast and can interfere with readability and word identification.[75] I replaced the underscores with checkboxes and added a vertical rule between the columns to make the visual hierarchy more apparent.

The new form can be filled out electronically or printed and filled in by hand. Since most of the staff engineers do not have laptops, both options were necessary.

---

[72] Riley and Mackiewicz, *Visual Composing*, 9.
[73] Ibid., 10.
[74] Ibid., 23.
[75] Ibid., 10.

PROJECT KICK-OFF MEETING CHECK LIST

PROJECT: _____ KICK-OFF DATE: _____

ARCHITECT: _____

In order to begin work on a new project, Robert L. Miller Associates must be in receipt of the following information:

IMMEDIATELY

_____ DIMENSIONED FLOOR PLAN WITH GRID DESIGNATIONS

_____ TYPICAL ARCHITECTURAL WALL SECTIONS

_____ OUTLINE SPECIFICATION, IF ONE HAS BEEN PREPARED

_____ BUILDING LOCATION:

_____

_____ BUILDING CODE: _____

_____ ROOF & FLOOR DATUM ELEVATIONS

_____ SIZE & LOCATION OF ROOF SCREENS

_____

_____ WIDE FLANGE OR TUBE COLUMNS?:

_____

_____ ROLLED SECTIONS OR JOINT GIRDERS?: _____

_____ EXPANSION JOINTS?: _____

_____ "X" BRACING OR RIGID FRAME?

_____

_____ WAS SNOW DRIFTING ON EXISTING ROOF DISCUSSED? _____

COMMENTS: _____

_____

_____ TYPE OF ROOF:

_____

_____ FACTORY MUTUAL REQUIREMENTS?:

_____

WITHIN 7 DAYS OF PROJECT KICK-OFF

_____ ARCHITECTURAL ELEVATIONS

_____ FINAL GRADING PLAN

_____ SOILS REPORT

_____ BUILDING CLEAR HEIGHT _____

_____ TRENCH OR CONVENTIONAL EXTERIOR FTGS?:

_____

_____ VERIFIED WITH BUILDING DEPT?: _____

_____ INTERIOR FTGS: ROUND OR SQUARE? _____

REINFORCED? _____

_____ IF PRECAST WALLS, WHO IS THE MANUFACTURER?: _____

_____ ROOF DRAIN LOCATIONS, AND

_____ ROOF DRAIN SLOPES

_____ RACK LAYOUT & LOADS

_____ ANTICIPATED SLAB ON-GRADE LOADING

_____ GENERAL CONTRACTOR:

_____

_____ SIZE, WEIGHT & LOCATION OF ROOFTOP MECH. EQUIPMENT:

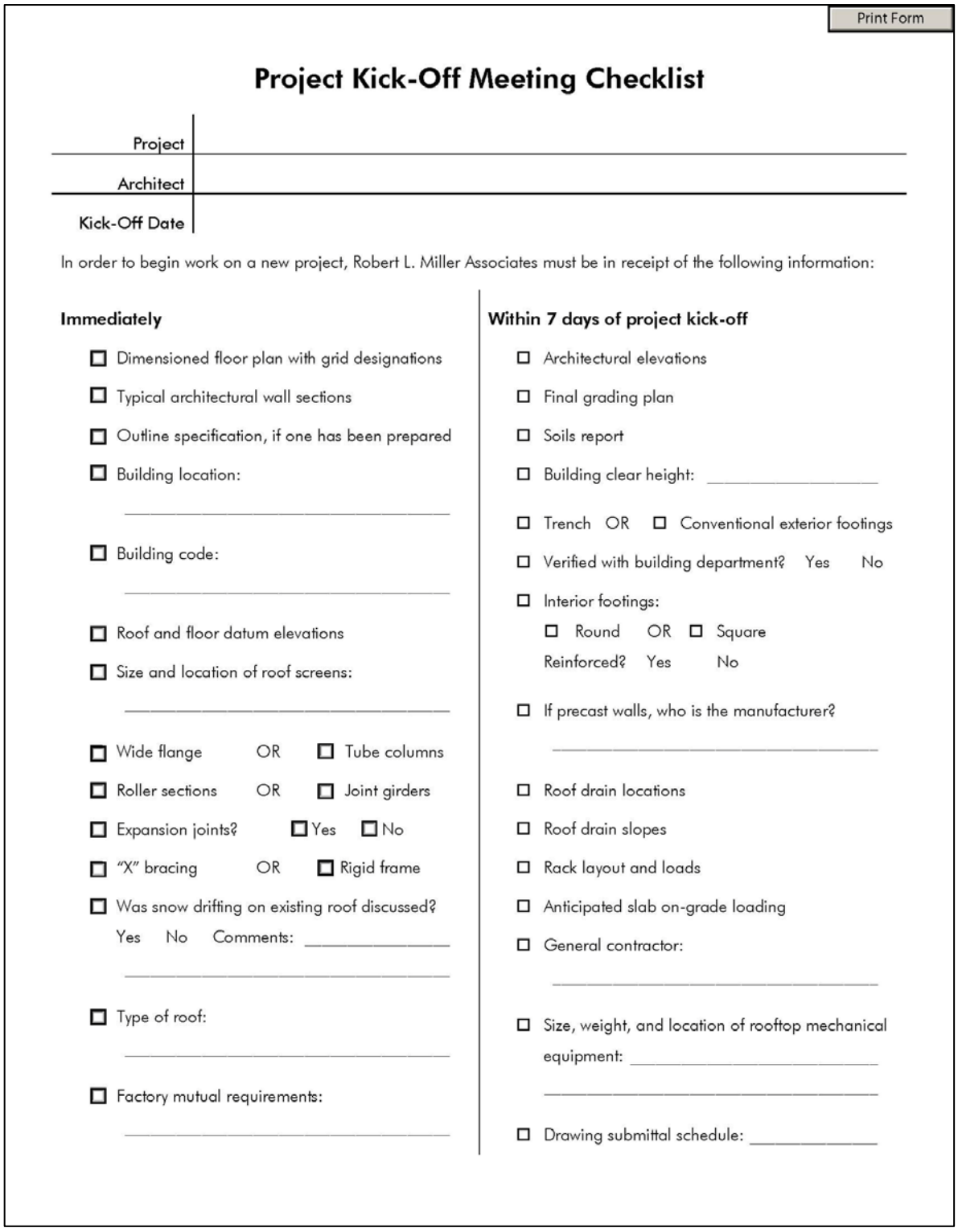
_____

_____ DRAWING SUBMITTAL SCHEDULE:

_____

_____

Figure 7. Original project kick-off meeting checklist



Print Form

Project Kick-Off Meeting Checklist

Project _____

Architect _____

Kick-Off Date

In order to begin work on a new project, Robert L. Miller Associates must be in receipt of the following information:

Immediately

- ☐ Dimensioned floor plan with grid designations
- ☐ Typical architectural wall sections
- ☐ Outline specification, if one has been prepared
- ☐ Building location: _____
- ☐ Building code: _____
- ☐ Roof and floor datum elevations
- ☐ Size and location of roof screens: _____
- ☐ Wide flange OR ☐ Tube columns
- ☐ Roller sections OR ☐ Joint girders
- ☐ Expansion joints? ☐ Yes ☐ No
- ☐ "X" bracing OR ☐ Rigid frame
- ☐ Was snow drifting on existing roof discussed? Yes No Comments: _____
- ☐ Type of roof: _____
- ☐ Factory mutual requirements: _____

Within 7 days of project kick-off

- ☐ Architectural elevations
- ☐ Final grading plan
- ☐ Soils report
- ☐ Building clear height: _____
- ☐ Trench OR ☐ Conventional exterior footings
- ☐ Verified with building department? Yes No
- ☐ Interior footings:
  - ☐ Round OR ☐ Square
  - Reinforced? Yes No
- ☐ If precast walls, who is the manufacturer? _____
- ☐ Roof drain locations
- ☐ Roof drain slopes
- ☐ Rack layout and loads
- ☐ Anticipated slab on-grade loading
- ☐ General contractor: _____
- ☐ Size, weight, and location of rooftop mechanical equipment: _____
- ☐ Drawing submittal schedule: _____

Figure 8. Redesigned project kick-off meeting checklist

Future Work

The immediate next step for this project is for Brian and Bob to continue to populate the wiki with content. Once the amount of content reaches a critical mass, the day-to-day uses of the system can be explored. As noted earlier, it is not sufficient for the wiki to exist; it must also be used.

In addition, even if the system is used as we hope, I realize that it alone cannot be the end-all solution to managing knowledge in the firm. In *The Fifth Discipline*, Peter M. Senge quotes Anne Murray Allen of Hewlett-Packard (HP), who came to a similar realization:

> "At HP, I got involved in strategy, and especially the strategic role of IT, which naturally led into knowledge management…. But all the 'lessons learned' databases and the like never seemed to me to be very high-leverage, nor the whole idea that knowledge is somehow floating all around us and all we have to do is capture and codify it. That notion has become a lot less popular now because companies have spent a lot of money on 'knowledge management systems' and don't have a lot to show for it."[76]

Allen's solution to this problem came from recognizing that "knowledge is social."[77] She explains:

> "To manage knowledge you need to address collaboration and tools that help people collaborate. Today, much of our work is on knowledge networks, which

---

[76] Senge, *The Fifth Discipline*, 270.
[77] Ibid.

> we also call networks of collaboration: how people work together to create value and to create new sources of value. This is a very organic process, but there are ways to understand it and ways to help rather than hinder it."[78]

Although we started this wiki by prepopulating it with content from Brian and Bob, and we will continue to do so, it will live on only through collaboration. Not only do the other engineers need to use it, but they must also be able to contribute their own content, which will help them reflect on their work. Allen worked with a social network researcher, Dennis Sandow, who states his vision for KM networks: "My real aim was to help people reflect on how they did their work. People are naturally interested in understanding how their own work occurs and explaining it to others…. Over time we've learned that networks of knowledge expand and become stronger through reflection."[79]

As we begin involving the rest of the engineers with the wiki, we need to help them see the benefits they will experience by using and contributing to it. Some of these benefits—such as reflection—are intrinsic. We may also be able to add extrinsic benefits, such as rewards for adding or updating content. Ideally, along the way, once-tacit knowledge will be shared, and innovation will result.

Senge states, "Tools are crucial to any deep learning process. Buckminster Fuller used to say that 'you cannot change how someone thinks,' but you can give them a tool 'the use of which leads them to think differently.'"[80] My hope is that this wiki is such a tool.

---

[78] Ibid.
[79] Ibid.
[80] Ibid., 286.

References

Brown, John Seely, and Paul Duguid. *The Social Life of Information*. 1st ed. Harvard Business Review Press, 2000.

"Confluence - Pricing", n.d. http://www.atlassian.com/software/confluence/pricing.jsp.

"Confluence Licensing | Atlassian", n.d. http://www.atlassian.com/licensing/confluence.

"Content and Social Collaboration Software | Atlassian", n.d. http://www.atlassian.com/software/confluence/overview.

"Create and Edit a Wiki - SharePoint Foundation - Office.com", n.d. http://office.microsoft.com/en-us/sharepoint-foundation-help/create-and-edit-a-wiki-HA010378208.aspx.

"Creating a FAQ Application (DevGuide.FAQTutorial) - XWiki", n.d. http://platform.xwiki.org/xwiki/bin/view/DevGuide/FAQTutorial.

Dave, Bhargav, and Lauri Koskela. "Collaborative Knowledge Management — A Construction Case Study." *Automation in Construction* 18, no. 7 (November 2009): 894–902.

"Document Templates & Forms (Features.Forms) - XWiki", n.d. http://platform.xwiki.org/xwiki/bin/view/Features/Forms.

Egbu, C O. "Managing Knowledge and Intellectual Capital for Improved Organizational Innovations in the Construction Industry: An Examination of Critical Success Factors." *Engineering, Construction and Architectural Management* 11 (2004): 301–315.

"Export Your SharePoint Wiki", May 14, 2009. https://tlt.stonybrook.edu/FacultyServices/Software/Documents/Export_SharePoint_Wiki.pdf.

"Exports (Features.Exports) - XWiki", n.d. http://platform.xwiki.org/xwiki/bin/view/Features/Exports.

"Full Features List — Confluence", n.d. http://www.atlassian.com/software/confluence/full-features.jsp#search-feature.

"How to Export All Files from a Sharepoint Site?", n.d. http://social.technet.microsoft.com/forums/en-US/sharepointgeneral/thread/5a599238-cc2e-47ab-9a9f-347abd3fc5fd.

"Installation (AdminGuide.Installation) - XWiki", n.d. http://platform.xwiki.org/xwiki/bin/view/AdminGuide/Installation#HPrerequisites.

"Installing Confluence on Windows - Confluence 4.1 - Atlassian Documentation - Confluence", n.d. http://confluence.atlassian.com/display/DOC/Installing+Confluence+on+Windows.

"Introduction to SharePoint Master Pages - SharePoint Designer - Office.com", n.d. http://office.microsoft.com/en-us/sharepoint-designer-help/introduction-to-sharepoint-master-pages-HA102019628.aspx?CTT=1.

Kaptelinin, Victor, and Bonnie A Nardi. *Acting with Technology: Activity Theory and Interaction Design*. Cambridge, Mass.: MIT Press, 2006.

"License (Main.License) - XWiki.org", n.d. http://www.xwiki.org/xwiki/bin/view/Main/License.

Lidwell, William, Kritina Holden, and Jill Butler. *Universal Principles of Design*. Berverly, Mass.: Rockport, 2010.

Lutters, Wayne G., Mark S. Ackerman, and Xiaomu Zhou. "Group Information Management." In *Personal Information Management*, edited by Jaime Teevan and William P Jones, 236–248. Seattle: University of Washington Press, 2007.

"New Features in Confluence Wiki | Atlassian", n.d. http://www.atlassian.com/software/confluence/whats-new.jsp.

Ping Tserng, H., and Yu-Cheng Lin. "Developing an Activity-based Knowledge Management System for Contractors." *Automation in Construction* 13, no. 6 (November 2004): 781–802.

Project Management Institute Staff. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. Exton: Project Management Institute, 2008. http://encompass.library.cornell.edu/cgi-bin/checkIP.cgi?access=gateway_standard%26url=http://techbus.safaribooksonline.com/9781933890517.

Riley, Kathryn Louise, and Jo Mackiewicz. *Visual Composing : Document Design for Print and Digital Media*. Boston: Prentice Hall, 2011.

Rockley, Ann, Pamela Kostur, and Steve Manning. *Managing Enterprise Content : a Unified Content Strategy*. Indianapolis, Ind.: New Riders, 2003.

Rubin, Jeffrey, and Dana Chisnell. *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*. 2nd ed. Indianapolis, Ind.: Wiley Publishing, Inc., 2008. http://www.books24x7.com/marc.asp?bookid=25203.

"Scroll Wiki Forms - Atlassian Plugin Exchange", n.d. https://plugins.atlassian.com/plugins/com.k15t.scroll.scroll-forms.

"Search Application (Extension.Search Application) - XWiki", n.d. http://extensions.xwiki.org/xwiki/bin/view/Extension/Search+Application#HDefaultSearchusingLucene.

"Searching Confluence - Confluence 4.1 - Atlassian Documentation - Confluence", n.d. http://confluence.atlassian.com/display/DOC/Searching+Confluence.

Senge, Peter M. *The Fifth Discipline: The Art and Practice of the Learning Organization*. New York, N.Y.; London: Currency Doubleday, 2006.

Spinuzzi, Clay. *Tracing Genres Through Organizations : a Sociocultural Approach to Information Design*. Cambridge, Mass.: MIT Press, 2003.

Sunstein, Cass R. *Infotopia: How Many Minds Produce Knowledge*. Oxford: Oxford University Press, 2006.

"Support (Main.Support) - XWiki.org", n.d. http://www.xwiki.org/xwiki/bin/view/Main/Support.

Udeaja, C., J. Kamara, P. Carrillo, C. Anumba, N. Bouchlaghem, and H. Tan. "A Web-based Prototype for Live Capture and Reuse of Construction Project Knowledge." *Automation in Construction* 17 (2008): 839–851.

Vicente, Kim J. *The Human Factor : Revolutionizing the Way People Live with Technology*. New York: Routledge, 2006.

"WikiMatrix - Compare Them All", n.d. http://www.wikimatrix.org/.

"WikiMatrix / SharePoint Wiki Plus Features - Compare Them All", n.d. http://www.wikimatrix.org/show/SharePoint-Wiki-Plus.

Woo, J-H, M J Clayton, R E Johnson, B E Flores, and C. Ellis. "Dynamic Knowledge Map: Reusing Experts' Tacit Knowledge in the AEC Industry." *Automation in Construction* 13 (2004): 203–207.

"Working with Templates - Confluence 4.1 - Atlassian Documentation - Confluence", n.d. http://confluence.atlassian.com/display/DOC/Working+with+Templates.

Wright, Alex. *Glut: Mastering Information Through the Ages*. Washington, D.C.: Joseph Henry Press, 2007.

"XWiki — Open Source Wiki and Content-oriented Application Platform", n.d. http://www.xwiki.org.

"XWiki Support - XWiki.com", n.d. http://www.xwiki.com/xwiki/bin/view/Offer/Support?tab=first%2Dtabber%2Dheader4&tabberIds=first%2Dtabber%2Dheader#first-tabber-header4.

Appendix 1: Project Charter

Project Purpose and Justification

Robert L. Miller Associates is a small structural engineering firm located in Rolling Meadows, Ill. They provide structural design services for a wide range of building types. Robert "Bob" Miller founded the firm in 1984. In 2009, Brian Dekker joined the firm and is in the process of purchasing the firm from Bob, with full transition of ownership to be complete in 2012.

The firm currently consists of six engineers, including Bob, who is in the process of reducing his hours and will retire from day-to-day involvement in the firm by 2013. As the firm grows, Brian is concerned that the various engineers' differing styles will lead to an inconsistency in their work product and client experience. He is also eager to find a way to capture Bob's wisdom and practices before he retires.

Therefore, Brian has commissioned the creation of a digital office standards manual. The manual will facilitate consistent, high-quality work from each engineer and increase efficiency, saving time and money.

Project Objective

Create a digital office standards manual that records the procedures involved from beginning to end of a structural project, including communicating with clients, making calculations, completing drawings, and archiving files.

High-Level Requirements

1. The manual must be digital, enabling searching by keyword.
2. The manual should be organized by topic with cross-linking between topics as appropriate.
3. The manual must be easily editable by authorized users, without the need for specialized knowledge or software.
4. The manual should contain all necessary content, as specified by Brian.
5. The manual should be protected from access by unauthorized users.

Summary Milestone Schedule

Objectives, requirements, and schedule approved by client	September 15, 2011
Existing content delivered by client	September 15, 2011
Table of contents delivered by client	September 21, 2011
Deliverable format and hosting platform selected	October 1, 2011
Existing content catalogued and list of missing content delivered to client	October 15, 2011
Platform secured and access accounts created	November 1, 2011
Missing content delivered by client	November 15, 2011
Content templates completed	December 1, 2011
Content template approved by client	December 9, 2011
Content analysis complete	February 15, 2012
Content transferred to platform	March 1, 2012
Client feedback received	March 8, 2012
Edits completed	April 1, 2012
Client sign-off received	April 8, 2012

Appendix 2: Revised Schedule and Deliverables

Detailed Schedule (Erica)

Tasks	**Dates**
Determine objectives, requirements, and schedule	August 24-September 21
Review existing content	September 1-September 21
Explore wiki engine options	September 19-30
Finalize wiki evaluation criteria	September 21-30
Choose three to five wiki engines for evaluation	September 26-October 7
Assess selected wiki engines	October 3-7
Write wiki engine recommendation outline	October 10-13
Draft wiki engine recommendation report	October 14-21
Finalize wiki engine recommendation report	October 24-November 4
Install wiki engine and create access accounts	October 24-November 4
Upload and format table of contents, template, and "Preliminary Work" and "Analysis" sections	November 7-December 5
Make edits to table of contents, template, and "Preliminary Work" and "Analysis" sections	December 12-30
Format "Design: Steel"	January 2- 6
Make edits to "Design: Steel"	January 9-13
Format "Design: Concrete"	January 16-20
Make edits to "Design: Concrete"	January 23-27
Format "Design: Masonry"	January 30-February 3
Make edits to "Design: Masonry"	February 6-10
Format "Design: Wood"	February 13-17
Make edits to "Design: Wood"	February 20-24
Format "Drawing and Specs" and "Construction Administration"	February 27-March 2
Make edits to "Drawing and Specs" and "Construction Administration"	March 5-9
Develop interactive elements (if any)[81]	March 12-30
Review all sections for missing content, broken links, etc.	March 12-30
Make final edits requested by client	April 2-6

[81] We had originally envisioned the project kick-off meeting checklist being converted into an interactive web form, where some basic information about the project could be entered and a form with only the applicable fields would be generated. That form could then be printed or filled out electronically. We decided the time involved in creating this feature would not have a sufficient corresponding benefit.

Detailed Schedule (Brian)

Tasks	Dates
Create table of contents	September 1- 21
Gather content for "Preliminary Work" and "Analysis"	September 22-November 4
Assist as needed in installation of wiki engine	October 24-November 4
Review table of contents, template, and "Preliminary Work" and "Analysis" sections on wiki and request edits	December 5-9
Gather content for "Design: Steel"	December 12-30
Gather content for "Design: Concrete"	January 2-January 13
Review and request edits to "Design: Steel"	January 9-13
Gather content for "Design: Masonry"	January 16-27
Review and request edits to "Design: Concrete"	January 23-27
Gather content for "Design: Wood"	January 30-February 10
Review and request edits to "Design: Masonry"	February 6-10
Gather content for "Drawing and Specs" and "Construction Administration"	February 13-24
Review and request edits to "Design: Wood"	February 20-24
Gather content for interactive elements (if any)	February 27-March 9
Review and request edits to "Drawing and Specs" and "Construction Administration"	March 5-9
Review all content and request final edits	March 12-30

Major Deliverables

Deliverable	Date
Wiki engine installed and access accounts created	November 4, 2011
Table of contents, template, "Preliminary Work" and "Analysis" up on wiki (Fall semester deliverable)	December 5, 2011
"Design: Steel" up on wiki	January 13, 2012
"Design: Concrete" up on wiki	January 27, 2012
"Design: Masonry" up on wiki	February 10, 2012
"Design: Wood" up on wiki	February 24, 2012
"Drawing and Specs" and "Construction Administration" up on wiki	March 9, 2012
Interactive elements (if any) live	March 30, 2012
Final deliverable completed	April 6, 2012

Appendix 3: Fall Semester Activity Log

Date	Activity
8/24/11	• Drafted project charter
8/25/11	• Met with advisor to review project charter
9/1/11	• Met with Brian: o Reviewed project charter o Talked through Brian's first draft of table of contents o Reviewed existing office standards binder o Scheduled next meeting (9/21) o Decided on next steps (Brian to work on table of contents, Erica to review existing content (start with defining loads) and investigate wiki systems)
9/14/11	• Wrote feasibility study proposal outline • Reviewed *Architectural Office Standards and Practices: A Practical User's Guide*
9/19/11	• Reviewed existing files • Began looking at possible wiki engine options on wikimatrix.org
9/20/11	• Wrote feasibility study proposal • Created activity log • Created revised schedule and deliverables document
9/21/11	• Met with Brian: o Reviewed feasibility study proposal o Revised schedule and deliverables document o Divided up table of contents into six sections for deliverables o Got account on RLMA network to access files and existing information on SharePoint o Scheduled next meeting (10/12)
9/24/11	• Created spreadsheet to compare wiki engines • Did Google searches for recommended engines
9/26/11	• Presented project at group research meeting and requested suggestions on wiki engines that do OCR or utility for generating checklists (no recommendations received)
9/27/11	• Attempted to log in to RLMA server (server timed out) • Revised schedule • Created Google calendar for schedule • Emailed Brian
9/30/11	• Asked for wiki engine suggestions on Twitter (no recommendations received) • Logged into RLMA server and was able to access Email and Shared Folders, but could not access Internal Web Site • Used wikimatrix.org to compare available wiki engines, narrowing 103 options to 19 for further evaluation.
10/4/11	• Eliminated four softwares from contention, analyzed three other against the criteria • Still unable to log into SharePoint
10/8/11	• Reviewed remaining 12 softwares, eliminating nine and analyzing three against

	the criteria • Still unable to log into SharePoint so assessed it based on info I could find online
10/12	• Met with Brian o Decided on Confluence as the frontrunner alternative to SharePoint o Will compare template capabilities between Confluence and SharePoint o Will look into how Confluence tracks users and copying and pasting images o Will register for 30-day hosted trial to test capabilities o Brian will have template by the 18th, aim to have content for first two sections by 11/3 (leaves on vacation on 11/4), if not then, by 11/21 o Scheduled installation for 10/27 at 5 p.m.
10/15	• Wrote feasibility study progress report
10/18	• Started feasibility report draft • Researched questions about Confluence from 10/12 meeting o Definition of user o Terms of license o Pasting images from clipboard • Looked at database requirements (recommend PostgreSQL, instructions at http://confluence.atlassian.com/display/DOC/Database+Setup+For+Any+External+Database and http://confluence.atlassian.com/display/DOC/Database+Setup+for+PostgreSQL) • Registered for a 30-day trial of the hosted version so we can test it out o Created a test page using a preinstalled template (meeting notes) o Was able to copy and paste an image directly into the page o Installed Forms plugin but didn't try it yet
10/19/11	• Finished feasibility study draft
10/22/11	• Experimented with templates and created three options • Learned Scroll Wiki Forms plugin is not free after all (costs additional $10 a year), but it's probably not necessary
10/26/11	• Looked up online/PDF access options for building codes and standards
10/27/11	• Met with Brian and Tom (staff engineer who maintains the server) and showed them the templates. They're having server problems, so rescheduled the install to next Tuesday, 11/1. • Tom asked about viewing the wiki mark-up directly. Discovered that 4.0 version doesn't have a built-in wiki mark-up editor anymore. Some users are upset about this (c.f. http://confluence.atlassian.com/display/DOC/Confluence+4.0+Editor+-+What%27s+Changed+for+Wiki+Markup+Users) so it's possible that could change in the future. • Looked into how to make creating a page from a template easier: http://confluence.atlassian.com/display/CONFEVAL/Create+a+Page+Using+a+Default+Template • Learned more about using form fields in templates: http://confluence.atlassian.com/display/DOC/Creating+a+Page+using+a+Template • Created a revised template with simpler looking tables
10/28/11	• Agreed on fee and put together payment schedule

	• Looked for way to add instructions to the template that wouldn't appear on the final page, but didn't find any. Not sure instructions are needed anyway. • Made a couple new templates (10-28 Revised Standard and 10-28 Revised Standard 2) that don't have any prefilled content that would need to be deleted. One has headings for the image boxes; the other just has empty boxes.
11/1/11	• Went into office in hopes of installing Confluence, but server had crashed when Tom tried to resize the operating system partition.
11/3/11	• Emailed Tom for server update
11/7/11	• Emailed Tom for server update and scheduled install for 11/8/11
11/8/11	• Met with Tom to install Confluence o Tom had already installed PostgreSQL database and downloaded Confluence o Assigned database settings o Bought Confluence license o Created accounts for Tom, Brian, and me o Added Office Standards as a new space on Confluence o Set my laptop up for VPN access o Changed home wireless router IP address to eliminate conflict with VPN
11/13/11	• Exported content from demo site • Tried to add Advanced Macros plug-in and got this message: "The system attempted to connect to http://localhost:8090/rest/plugins/1.0/pending/f035e6fe-78fd-4f9e-afea-318bf063eaab to retrieve the status of your operation, but failed. This may indicate a problem with the base URL configuration of your instance. Try refreshing the page to see if the operation completed, and see the UPM documentation (http://confluence.atlassian.com/display/UPM/Diagnosing+base+URL+configuration+problems) for more details about this error." Plug-in appears in list, however, so seems successful. Then hit "upgrade all" for plug-ins listed. Same message occurred, but seems to have been successful. • Clicked through all menus and settings on install to acclimate myself o Changed site name (title displayed in browser) o Updated all plug-ins (see above) o Interestingly, English – U.K. language pack is installed, and U.S. pack is only 20% complete. Leaving as-is for now. o Imported default templates o Configured Whitelist to allow all URLs o Need to ask Tom for mail server info to allow system to email updates o Edited permissions to allow users to attach files and update status and allow non-logged-in users to view content o Uploaded logo
11/14/11	• Created three pages to see how navigation looks. It appears to sort alphabetically, so added numbers before section titles. • Met with advisor, scheduled day to show first semester progress (12/12)
11/17/11	• Gave anonymous users ability to view Office Standards space (since it's impossible to access without already being logged into the RLMA server)

	• Changed default start page from Dashboard to Office Standards home • Edited welcome message • Figured out how to change order of pages in navigation sidebar using Browse-Pages-Tree tab
11/18/11	• Added Standards template (from demo site) • Added the rest of the five section pages • Pasted content into Load Determination page from Office Standards 2011 Word doc. Equations did not copy over (need to figure out how to recreate or do screenshots), and images need to be copied one by one. Need to figure out if this content should all be on one page or broken up.
11/29/11	• Reviewed activity log and deadlines o Original deadline for this semester's deliverable was 12/5, but not meeting with advisor until 12/12, so counting that as the deadline o Still need additional content from Brian for Project Kick-Off and Analysis • Figured out how to use http://www.codecogs.com/latex/eqneditor.php to generate gifs of equations to insert into pages. Can copy equation from Word document, paste into text box on site, download gif, and insert into wiki page. • Broke Load Determination page into three pages: Load Determination, Design Loads, and Snow Loads. If more content is added, it will probably be necessary to add additional subpages. • Finished creating starter pages for Brian to populate in Project Kick-Off and Analysis • Emailed Brian
11/30/11	• Phone call with Brian o He has ordered the code references we need. o He's working on the content and will aim to have it in the system by next Wednesday (12/7).
12/9/11	• Phone call with Brian o Has some content added o Will add remaining content on Saturday
12/11/11	• Redesigned Project Kick-off Checklist • Checked for missing content • Made design tweaks • Discovered that images stay attached to the page where they were originally pasted, even if they are deleted and then used on a different page. Can change which page they're attached to by clicking Properties and move to correct page.
12/12/11	• Semester progress meeting with advisor

Appendix 4: Wiki Table of Contents (Proposed)

Note: To save space, I did not list the names of all of the sub-pages.

1. Preliminary Work
    1. Kick-off Meeting Checklist
    2. Load Determination
    3. Structural System Selection
    4. Foundation System Selection
    5. Geotechnical Reports
2. Analysis
    1. Software
3. Design
    1. Steel
       (with 42 sub-pages)
    2. Concrete
       (with 37 sub-pages)
    3. Masonry
       (with 31 sub-pages)
    4. Wood
       (with 21 sub-pages)
    5. Cold-Formed Steel
       (with 1 sub-page)
    6. Steel Joists and Joist Girders
       (with 5 sub-pages)
    7. Metal, Non-Composite and Composite Deck
       (with 4 sub-pages)
    8. Special Topics
       (with 1 sub-page)
    9. Pre-engineered Metal Buildings
4. Drawings and Specs
    1. Standard Details
    2. Go-by Drawings
    3. General Notes
    4. Master Spec
    5. Calculation Format
5. Construction Administration
    1. Shop Drawing Review
    2. Field Reports
    3. Sharing CAD Files