ABSTRACT

MY TECHNICAL WRITING INTERNSHIP AT BLUESPRING SOFTWARE

by Angela Shealy

The purpose of this report is to describe and reflect upon my technical writing internship at Bluespring Software, Inc. My internship enabled me to apply my technical writing skills learned and honed throughout the MTSC program in a professional setting and to products and applications used in the business arena.

This report is comprised of four chapters. Chapter 1 provides a basic background on Bluespring Software as a company and the initial framework of my internship report. Chapter 2 introduces my internship goals and the projects on which I participated while completing those goals. Chapter 3 examines the major project I completed for Bluespring Software—the Priority CS® Administrative Management Console (AMC) training manual. Finally, chapter 4 discusses the projects reviewed in Chapter 3, as well as Bluespring Software’s satisfaction with my performance and work produced throughout internship.
MY TECHNICAL WRITING INTERNSHIP AT BLUESPRING SOFTWARE

An Internship Report

Submitted to the

Faculty of Miami University

in partial fulfillment of

the requirements for the degree of

Master of Technical and Scientific Communication

Department of English

by

Angela Lynn Shealy

Miami University

Oxford, Ohio

2010

Advisor_____________________________

Dr. W. Michele Simmons

Reader_____________________________

Dr. Hugh Morgan

Reader_____________________________

Dr. Janel M. Bloch
# Table of Contents

Chapter 1: Introduction to my Internship Experience at Bluespring Software

- Bluespring Software, Inc. ................................................................. 1
- My Introduction to Bluespring Software ............................................. 3
- Corporate Culture at Bluespring Software ......................................... 4
- My Internship Report ........................................................................ 5

Chapter 2: Overview of My Internship.................................................. 6

- Internship Goals ................................................................................ 6
- Table 1: Revised Employee Development Plan .................................... 7
- Initial Projects at Bluespring Software ................................................ 8
  - Administrative Management Console (AMC) Configurator Training Manual v3.1  8
  - csrWeb v3.1 eLearning Application .................................................. 9
- Bluespring Software Website ................................................................. 10
- Subsequent Projects at Bluespring Software ....................................... 11
  - The AMC Installation Guide v3.2 ...................................................... 11
  - Microsoft SharePoint Team Services Training ................................ 11
  - Marketing Collateral ........................................................................ 13
- Conclusion ......................................................................................... 13

Chapter 3: Priority CS® Administrative Management Console: Training Manual ............. 15

- Overview of My Major Project ............................................................ 15
- Training Manual Design .................................................................... 16
  - Requirements for the Class ............................................................. 17
- Usability Testing ................................................................................ 19
  - Usability Tester Selection ............................................................... 22
  - Bluespring Software’s Usability Testing Solution .............................. 23
  - Evaluation of Usability Testing Results and Revisions ..................... 25
- Conclusion ......................................................................................... 26

Chapter 4: Analysis of Methodologies Implemented: How MTSC Shaped My Internship ...... 27

- Defining the Problem ......................................................................... 27
- Designing the Solution ....................................................................... 28
- Testing the Solution ........................................................................... 29
- Implementing the Solution .................................................................. 30
- Evaluating the Solution ...................................................................... 31
- Conclusion ......................................................................................... 31

References .......................................................................................... 34
List of Tables

Table 1: Revised Employee Development Plan................................................................. 7
List of Figures

Figure 3-1: Excerpt from AMC Training Manual ................................................................. 20
Chapter 1: Introduction to my Internship Experience at Bluespring Software

I accepted an internship at Bluespring Software, Inc., as a user services analyst in August of 2003. I was very much looking forward to the internship. Their only current technical writer—Kathryn Rudolph—and I had attended the Master of Technical and Scientific Communication (MTSC) program together, and we had worked well together on projects in the past. Also, it was made very clear throughout my interview process that the company recognized the importance of technical writing, and it was making a strong effort to both maintain existing documentation and establish new documentation for its evolving software program. In addition, I was told that if the company were pleased with my progress during the internship, it would consider hiring me on full-time, which was certainly a major goal of mine throughout the internship. Indeed, at the end of the internship, I was hired into the company as a full-time technical writer and technical trainer, where I worked until I left the company for another opportunity in June of 2004.

Bluespring Software, Inc.

Bluespring Software, Inc., is a software company located in Cincinnati, Ohio, that developed an application specifically aimed at performing pre-sale opportunity assessments of non-standard requests (requests outside the scope of traditional business offerings) in the telecommunications industry. The application, Opportunity Feasibility Management (OFM), evaluates non-standard requests on an individual case basis to determine the feasibility and profitability of each using a business process management (BPM) strategy. BPM is the generic name for a management tool that allows companies to automate business processes by monitoring how they are performing against business plans, holding individuals accountable for their performances within the organization, and creating automatic and manual tasks to complete simple and complex business procedures. Bluespring Software’s first BPM product was Priority CS®, which later evolved and became an integral piece of Bluespring Software’s current OFM technology. When I joined Bluespring Software, its mission statement was printed in my Bluespring Software Associate Handbook as follows:
Bluespring Software’s mission is to be the OSS [Open Source Software] and Billing Systems software provider of choice for the telecommunications industry. To be recognized for distinctive products and services developed to exacting customer criteria. To be recognized throughout the industry by suppliers and competitors, for our innovativeness, world-class execution, enlightened leadership, and ethical business practices. To be trusted by the customers we serve and the communities in which we operate. To be seen as a value-added software provider, bringing something of genuine distinctiveness to the marketplace. And finally, to be revered by Bluespring Software associates and their families for providing challenging, developmental and substantially rewarding career opportunities.

Toward the end of my internship, Bluespring Software was broadening its focus to include customers outside of the telecommunications industry who might also have business needs for BPM software. For example, Bluespring Software was offering its OFM technology to two very different types of solutions providers near the conclusion of my internship: one was a security solutions provider and the other was a property management solutions provider. This new, wider focus emerged primarily because Bluespring was struggling to acquire the amount of business it desired from specializing in the telecommunications industry.

The company was founded in 1997 as Software Management Group (SMG). In 1999, SMG approached the Greater Cincinnati Venture Association (GCVA) and the Ohio Venture Association (OVA) to try to gain venture capital for their emerging products. In 2000, SMG signed its first two clients and changed its name to Bluespring Software, Inc. In 2002, Bluespring Software’s business concentration shifted to OFM – a service that incorporates Bluespring Software’s existing Priority CS® tool with the business need to automate and document prospective growth areas, improve business process efficiency, and better meet the needs of current and potential customers. This allowed Bluespring Software to offer its products to a broader market that could include customers outside the telecommunications industry.

Bluespring Software had faced some turmoil in its early years, including three rounds of layoffs in 2001. Throughout my internship, the number of employees hovered around 20, with both turnovers and gains in this area. While the company had one solid customer (AT&T Canada) when I joined, there was constantly an underlying uneasiness among most employees regarding the stability and longevity of the company. In my 10 months with the company, we experienced another layoff—which jeopardized the morale of some of the most dedicated employees. Most
employees learned to channel this uneasiness and concern into motivation for developing and maintaining a quality product. All of the employees believed strongly in Bluespring Software’s ability to achieve its goal of becoming the leading provider of OSS and Billing Systems software for the telecommunications industry, if it were given the chance. Bluespring Software employees, from upper-management to developers to technical writers, fought hard to keep their company afloat, and their persistence paid off; when I left the company in June 2004, Bluespring Software had signed on another client and had several more prospective clients in its pipeline.

My Introduction to Bluespring Software

When I joined Bluespring Software, technical communication had become a priority. In previous layoffs, the entire technical writing team had been terminated, and much of the documentation was either completely out-of-date, or written by busy developers who whose priorities centered around developing the application—not documentation. Kathryn Rudolph joined Bluespring Software in 2002 as an intern and was eventually hired on as a full-time employee. She had begun the daunting task of revitalizing Bluespring Software’s documentation; however, both she and Karl Treier, the CTO, agreed that another technical writer was a necessity. When I joined the documentation team, Kayte showed me the ropes, and together we worked to develop quality documentation for Bluespring Software’s products and marketing materials. While we were generally in charge of separate documentation projects, we supported each other when needed. Eventually, Kayte took on more of the web-based projects, such developing Bluespring Software’s website and creating an eLearning applications for Priority CS®, whereas I was put in charge of developing the in-class training materials and helping with marketing materials.

Several weeks into my internship, I learned Bluespring Software was also in need of a trainer. The CTO asked me if I were interested in pursuing that role within the company. I would still be considered a technical writer in the documentation department, but I would also have the added responsibility of developing Priority CS® training curriculum and conducting in-class training sessions for both potential and existing customers. This fit well with my existing project, and the CTO, who had 12 years of training experience, had evaluated my presentation skills and felt I would excel as a trainer. To develop my training skills, I enrolled in the CompTIA Certified Technical Trainer (CTT+) certification program and passed with flying colors. I achieved my
CTT+ certification in January and officially became Bluespring Software’s sole trainer. Because we had such a limited clientele at the time, I continued to focus on developing a strong training manual and helped with a variety of other documentation projects.

**Corporate Culture at Bluespring Software**

Bluespring Software employees—from the CEO to managers to staff—had a unique camaraderie, which was evident in the company’s culture. In many corporate cultures, employee career levels and hierarchies are defined by distinct, albeit invisible, boundaries. Although the CEO and some upper-level managers were involved in decision-making strategies regarding the layoffs that occurred, company budget cuts (including employee cuts) were a direct result of lack of funding by investors, and not directly attributed to any individual who worked at Bluespring Software. As such, Bluespring Software employees at almost every career level were encouraged to interact on both a professional and casual level with others. It was important to the office environment that people “got along” and worked well together. Employees from different departments and career levels often worked together as partners on projects, and everyone’s input was considered equally important.

Another unique element contributing to Bluespring Software’s corporate culture was the expectation that employees should have periods of downtime throughout the day. For example, the company purchased a foosball table, which was kept in a conference room located toward the back of the office. Bluespring Software employees often took 10 to 20 minute “foosball breaks,” to play foosball games (and relieve stress). All employees were encouraged by their peers to participate in this past-time, even if only to observe. This specific example stands out to me because I believe it reveals Bluespring Software’s attempt to balance work with a sense of fun. Employees were expected to take their jobs and their work seriously, but the company also recognized the value of short breaks that allowed employees to de-stress and come back to work revitalized. I believe this environment truly made the workplace more productive.

Gareth Morgan’s book “Images of Organization,” provides an illustration of “organizations as organisms,” in which he likens organizations to organisms: like biological organisms, individuals and groups operate most effectively when their needs (physiological, safety, love/belonging,
esteem, self actualization) are being met. Bluespring Software provided an environment where most employees’ needs were sufficiently met, thus enabling them to engage in their work and achieve mutual goals. For example, foosball breaks allowed employees to relieve stress while building camaraderie among the group, which then translated into a sense of belonging and self-esteem that enabled groups to work together enthusiastically toward shared objectives. I believe Bluespring Software’s unique culture was partly due to the small size of the company, and partly due to the turbulent experiences that many employees underwent as a result of seeing the company downsized so drastically. The remaining employees’ sense of belonging, loyalty to the company, and their belief in the products formed a close-knit community.

**My Internship Report**

My internship report provides a comprehensive synopsis of my internship experience with Bluespring Software, with an in-depth analysis of the major project I completed and the lessons I learned. Chapter 2: Overview of My Internship introduces my experience at Bluespring Software and discusses the projects in which I participated or owned. My major project, the Priority CS® Administrative Management Console (AMC) training manual, is discussed in more detail in Chapter 3. Chapter 4 provides an analysis of my work, methodology, and internal processes for completing the project discussed in Chapter 3, including how my MTSC training shaped my internship. It also includes a discussion on the company’s overall satisfaction with the deliverable resulting from the project (training manual) and its value to current and prospective clients. As a side note, in order to avoid potential infringements upon proprietary rights, I did not include actual work products in my internship report. However, throughout the report, I provide examples and discussions of the work products and projects to which I contributed.
Chapter 2: Overview of My Internship

I was told at the beginning of my internship that I would be treated as a full-time employee, with the possibility that I would become fully-employed at the conclusion of my internship. Because Bluespring Software was a small company, all employees were expected to wear several different “hats” and contribute multiple talents to their work at the company. As I quickly found out, interns were no exception. During my first week at Bluespring Software, I was immediately given several projects. Personally, I found this diverse work environment stimulating, and I enjoyed being able to contribute to different areas of the company. I also found that honing skills in different areas of the company gave me a better “big-picture” understanding of Bluespring Software and its products, which in turn was reflected in my documentation. I was also able to better conceptualize the visions and goals of our company. I felt challenged and valued when I began my internship, which I’ve learned are both very important components to my happiness in my chosen career. Even before I joined the company as an official employee near the conclusion of the internship, I was juggling several projects that required different skills. This chapter contains an overview of the projects I worked on throughout my internship at Bluespring Software.

Internship Goals

When I joined Bluespring Software in October of 2003, I met with my supervisor, Mr. Treier, and together we developed four goals that I would be responsible for accomplishing during my internship. These goals were developed with my internship and employee development needs in mind, as well as Bluespring Software’s need for a successful training program. When we formulated the goals, we also mapped out a plan for achieving each goal and cited the specific criteria that must be met in order to attain that goal. The goals were documented in my Employee Development Plan (Bluespring’s employee performance planning and evaluation tool). Initially, we agreed upon the following four goals:

1. **Demonstrate competency in technical writing. Apply the techniques and skills learned in the MTSC program.** Bluespring was in need of a second technical writer in order to develop several instructional and training manuals for their custom applications, as well as support
the development of their website. Mr. Treier wanted to ensure that my technical writing skills were solid and that I would be an asset to the company.

2. **Demonstrate basic project management skills.** Bluespring wanted all of its employees to acquire and practice basic project management skills. Since the company was so small, it was important that employees knew how to manage their own responsibilities, as well as assist co-workers in various projects.

3. **Assimilate fundamental OFM product knowledge.** It was important for me to have a solid understanding of Bluespring’s products—especially OFM, which was the core of its services and products. This would give me necessary background upon which to build the technical documentation that I would be creating.

4. **Develop Macromedia Flash content creation skills.** During my internship, Bluespring was in the process of developing a new website. Bluespring had selected Macromedia Flash as a tool for web design, and Karl was hoping that I could contribute to the design of the website.

Several weeks after we completed my Employee Development Plan, business needs dictated that Kayte would be learning the Macromedia Flash product (referred to in the fourth goal), and that I would contribute elsewhere within the company. As a result, the fourth goal was removed, and three new goals were added. The table below shows how the new goals appeared in my revised Employee Development Plan.

### Table 1: Revised Employee Development Plan

<table>
<thead>
<tr>
<th>1.) Goal: Demonstrate competency in Technical Writing. Apply techniques and skills learned in MTSC program.</th>
<th>How will you achieve this goal?</th>
<th>How Will We Measure?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author, based upon knowledge gained from Subject Matter Experts identified by the CTO in the OFM Installation Guide. Create documentation that is easy to read and understand. Ensure consistency in style, voice, grammar, etc. Possibly create style guide?</td>
<td>Completion of PriorityCS Installation Guide that meets the requirements defined by the CTO by the end of the Internship period. Conduct user testing (internal or external) on user guide to determine its usability.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2.) Goal: Demonstrate basic project management skills.</th>
<th>How will you achieve this goal?</th>
<th>How Will We Measure?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete the 3.1 Configuration Analyst Training Guide by ensuring Subject Matter Experts deliver contributions on time.</td>
<td>Completion of the 3.1 Configuration Analyst Training Guide by October 6th.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.) Goal: Assimilate fundamental OFM product knowledge.</th>
<th>How will you achieve this goal?</th>
<th>How Will We Measure?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create marketing content for the Bluespring Internet Site. Talk with SMEs and execs about the technicalities behind how the product works and what it does. Learn about the unique features of the OFM product and how it benefits customers.</td>
<td>Unaided content creation to specification given by Marketing Manager that meets with his approval. Create persuasive content that accurately and effectively describes OFM and emphasizes how it benefits customers.</td>
<td></td>
</tr>
</tbody>
</table>
4.) Goal:

How will you achieve this goal?

Expand organizational skills

Help locate and organize the “lost” documentation files/folders.

How Will We Measure?

Implementation of an organizational scheme. Store documents in an easy-to-navigate location.

5.) Goal:

How will you achieve this goal?

CTT+ Certification by 12/31/03

Pass the CTT+ certification exam and video

How Will We Measure?

Scores on the exam and video

6.) Goal:

How will you achieve this goal?

Internal demonstration of technical training skills

Design and present training material on SharePoint—teach different departments how SharePoint can meet their specific needs

How Will We Measure?

Departments effectively use SharePoint as a document management tool.

Throughout my internship, I was able to attain all of these goals, successfully meeting the expectations of my supervisor and other executives at the company.

**Initial Projects at Bluespring Software**

During my firsts few weeks at Bluespring Software, I was assigned primary responsibility for three projects, which are discussed below. As my internship progressed and the company became more comfortable with my skills, I was assigned several additional projects, which are reviewed in the subsequent section.

**Administrative Management Console (AMC) Configurator Training Manual v3.1**

I was assigned to work with Bluespring subject matter experts (SMEs) in OFM, developers, and programmers to complete the Administrative Management Console (AMC) Configurator Training Manual v3.1. The training manual was designed to teach new AMC users how to use Bluespring Software’s Priority CS® AMC, which was the business process management (BPM) tool used to configure the OFM application. The manual, which was to be used during a two week intensive course, contained 20 training topics covered in modules and in-class labs. It also contained the MS PowerPoint slides used during the instructor’s presentations. My major duties included editing and reformatting the modules, labs, and PowerPoint slides; testing the labs to ensure their accuracy; and working with SMEs, developers, and programmers to resolve any inconsistencies in the labs or modules. I also made the manuals easier-to-use by ensuring accuracy in the lab instructions, reorganizing some of the modules to provide information in a more chronological order, and eliminating wordiness throughout the modules and PowerPoint presentation. Chapter 3 describes in detail the development process used during the completion
of the AMC manuals, as well as the processes used to test the usability and accuracy of the manuals.

I completed the manuals by their assigned due date, and Mr. Treier ask me to give a presentation to a representative from Price Waterhouse Coopers (a consulting company considering purchasing the AMC) that discussed the benefits and importance of Bluespring Software’s AMC configurator training. The manuals were printed at Kinko’s and bound in official Bluespring Software training notebooks. Mr. Treier, my supervisor and CTO (who was also the then-trainer for the product), was very impressed with the final product, and he believed the training manuals would be effective and easy-to-use. He felt that the presentation was very effective and that it was good practice for becoming a trainer. At the conclusion of the meeting, the Price Waterhouse Coopers representative stated that he would recommend his company seriously consider purchasing the software.

**csrWeb v3.1 eLearning Application**
The csrWeb v3.1 eLearning course was a web-based training program designed to thoroughly educate users on the features and functions of csrWeb, a component of the AMC application. Typical users included individuals from the organization that had purchased the application who would be using csrWeb to complete their work. For example, developers, administrators, and programmers, as well as individuals with management responsibilities, were frequent users of the application and the target audience of csrWeb. The eLearning course contained interactive modules, such as quizzes and feedback components, which provided users with hands-on experiences using csrWeb. I worked in conjunction with Kayte Rudolph on this project. My primary responsibilities included creating interactive graphics and writing content for certain topics. Kayte and I collaborated by brainstorming ideas and dividing up content development. We also reviewed each others’ work. We grouped topics logically and chronologically, but allowed users to skip to different sections of the course, or to different topics, as needed. This enabled users to use the csrWeb as either a tutorial or as a reference guide/refresher course. We kept this same navigation structure consistent throughout the tutorial.
The csrWeb v3.1 eLearning application needed some revisions after its initial release, but the end-product was successful. This eLearning course was used by Bluespring Software’s largest client, who was very happy with the tool.

**Bluespring Software Website**

I was asked to create an educational section for Bluespring Software’s website. The educational section would briefly explain how the OFM application benefited customers, discuss both our on-site configurator training and our web-based eLearning course, reiterate the importance of training, and include a video about Bluespring Software’s OFM application. I was also responsible for ensuring consistency in layout and content and updating content and graphics throughout the entire website.

A couple of weeks into this project, the development of the website was transitioned to Devtopia, a professional website development firm that was hired to redesign and restructure our entire corporate site. Although I was somewhat disappointed that I wouldn’t get to have a more active role in the development of the educational portion of the website, the outsourcing of this project allowed me to focus my efforts on some of my more pressing responsibilities.

These initial three projects enabled me to address my first three goals, as follows:

1. **Demonstrate competency in technical writing. Apply techniques and skills learned in MTSC program.**
   By developing the AMC manual, assisting Kayte with the csrWeb v3.1 eLearning application, and creating a section of the website, the intent was to allow me to complete projects that specifically allowed me to show the skills I had developed during the MTSC program. These three projects allowed me to show both my instructional design and writing skills, as well as implement usability testing and publish the final product.

2. **Demonstrate basic project management skills.**
   By owning the AMC manual, I was able to continue to develop vital skills that would serve me throughout my professional career. While I had started to develop project management skills during my MTSC coursework, both when working individually and as a team, I was now learning how to manage a project in the “real-world”—a professional setting. The AMC manual enabled me to demonstrate that I could manage a project from cradle to grave, managing both my own time, as well as that of others during “sub-projects,” such as usability testing.
In addition, assisting with the eLearning application and website allowed me to experience working in a team setting in a professional environment. These two projects enabled me to fill supporting roles and learn more about Bluespring Software and my co-workers.

3. Assimilate fundamental OFM product knowledge.
Since Mr. Treier and the other Bluespring Software employees had very little time to teach me about their OFM products, as well as impart industry knowledge, I had to gain much of my OFM knowledge on my own. As such, I sifted through as much documentation as I could find and did my own “usability testing” of the products to which I had access (particularly the AMC software). My MTSC coursework had prepared me well for assimilating and employing new information quickly, which enabled me to accomplish this goal. The website project, in particular, helped me with this goal, as it required that I create persuasive, accurate, and effective content emphasizing the benefits of OFM.

Subsequent Projects at Bluespring Software

Halfway through my internship, I was asked to spearhead several additional technical communication projects, and each incorporated different aspects of technical writing and design.

The AMC Installation Guide v3.2
The AMC Installation Guide v3.2 was a comprehensive instructional manual that explained how to install Bluespring Software’s Priority CS® software on a company’s individual computers and its system-wide database. It included step-by-step procedures, scalability models, troubleshooting advice, and best practice guidelines designed to enable users to correctly install the components needed to access Priority CS®’s desktop and web applications. I was responsible for writing and designing the entire installation guide. To develop the guide, I worked frequently with SMEs to learn how to install the application myself. I translated the language that developers used to make the procedures understandable across a variety of audiences, without “dumbing down” the content. I organized the topics in a chronological order and provided headings to provide structure and forecasting. Because we were still building our audience base, it was important to make the documentation usable and readable across different industries and skills.

Microsoft SharePoint Team Services Training
During my internship, Bluespring Software adopted Microsoft SharePoint Team Services, a component of Windows 2003, as a tool for document storage and organization. The company
needed an online repository for its documentation that allowed for collaboration and sharing, and SharePoint was selected by the executive team for this purpose. Since no one in the company was familiar with this product, I was assigned to design and implement an internal training program that examined the unique features of SharePoint and to teach employees how to use the system to create, store, and manage documents. I was also responsible for creating a training manual that included modules, in-class labs, and presentation slides. This manual provided step-by-step instructions on using SharePoint, and engaged students in a series of labs progressively taught basic through advanced skills. I designed it to function either as a stand-alone document or to be used in conjunction with a training class.

In addition, I was given the opportunity to present a two-hour training session for each of Bluespring Software’s three major departments. The presentation consisted of both lecture and in-class labs, and it taught the basics of the SharePoint Team Services product.

Prior to the training presentations, Bluespring Software sent me to a train-the-trainer course in Columbus, Ohio, held by Langevin Learning Services and titled “Instructional Techniques for New Instructors.” The purpose of sending me to this course was twofold: my boss and company executives wanted to determine whether I had the capability, talent, and basic skills to become a good technical trainer, and they also wanted me to have the opportunity to ensure that I was comfortable with adding “training” to my primary job responsibilities. After taking the course, I felt confident that training fit well with my career path. I trained Bluespring Software’s employees how to use the SharePoint Team Services product, and eventually I became a certified technical trainer. This project was impactful for me because I found that I very much enjoyed training, and after the completion of my internship (and after being hired full-time by Bluespring Software), I became its sole trainer and was put in charge of its training program.

The Microsoft SharePoint Team Services project enabled me to address my last three goals, as follows:

4. **Expand organization skills.**

Microsoft SharePoint Team Services is an online document repository that allows users to share documents and collaborate on projects. Bluespring Software decided to migrate all of its remaining historical files to SharePoint, and then use the tool going forward as its primary
document repository. This allowed multiple users to access and update the same files. Because I was the first to learn the program (due to my training preparation), Mr. Treier asked me to set up the organization scheme, which would house thousands of documents. I consulted with the different areas of the business to determine the categories that should be included, and set up the initial structure. I asked representatives from each of the areas to ensure that the paths to their documents were intuitive and that crucial information could be found quickly easy. As folders and files were continuously added and updated, the paths changed and grew, but the organizational methodology and foundation were set by me at the beginning stages of the project.

5. CTT+ Certification by 12/31/03.
Since Bluespring Software was in need of a technical trainer when I started my internship, and since Mr. Treier thought that I had good presentation and communication skills, he suggested that I consider becoming a certified technical trainer. When I made the decision to pursue the certification, this goal was added to my Employee Development Plan. I studied for and took a written exam. In addition, I was required to submit a video-taped training session, and I used a SharePoint training session for my video-tape submission. I passed the exam before the end of December 2003 and became Bluespring Software’s technical trainer.

6. Internal demonstration of technical training skills.
As stated above, as part of the Microsoft SharePoint Team Services Training project, I designed and presented training material to teach several departments at Bluespring how to use the solution, both as a collaboration management solution and as a document storage repository. This training allowed me to achieve the sixth goal, which was to demonstrate my training skills to my supervisor (and his supervisors) to ensure that the company was comfortable with me taking on the role of technical trainer and providing product training to its customers.

Marketing Collateral
Because Bluespring Software’s website was being redesigned, I modified Bluespring Software’s print marketing materials using design features employed in the website to ensure consistency in our marketing collateral and build a clear relationship between our online and print documents.

Conclusion
My internship was a very successful experience. I received the opportunity to apply the concepts, techniques, and skills that I learned in the MTSC program to an actual employment situation. I think the most obvious proof of the success of my internship was that I was hired as a full-time employee! Essentially, the “real world” application of the skills I developed in the MTSC program resulted in a “real” job. I also developed a better understanding of concepts that are
often intangible before your first job, such as professionalism and corporate culture. Almost all of the experiences that I gained at Bluespring Software allowed me to evolve or refine skills that have benefited me thus far in my career. At Bluespring Software, my first—and most ambitious—assignment enabled me to manage a project and demonstrate the technical writing, teamwork, and project management skills that I had been developing in the MTSC program. Chapter 3 explains how I progressed through the project by engaging (and continuing to polish) these skills.
Chapter 3: Priority CS® Administrative Management Console: Training Manual

My major project during my internship at Bluespring Software was the continued development of the Administrative Management Console (AMC) training manual. The manual was originally developed by software developers at Bluespring Software, and then handed off to a technical writer who had been brought in to help with software documentation. Since the manual’s last revision, the software underwent several changes and additions, making much of the documentation, instructions, and screen captures obsolete. In addition, some of the manual’s original content was never fully developed, and many portions had not been tested for usability. However, because much of the training documentation had to be rewritten, results from the original usability testing also needed to be retested.

Overview of My Major Project

Essentially, the entire AMC training manual needed to be re-evaluated for its content accuracy, instructional suitability, and usability. It then needed to be updated to account for software changes, content gaps or inaccuracies, and tested for usability. (Usability testing is discussed in detail in the “Usability Testing” section of this chapter.) When I joined Bluespring Software and took ownership of the manual, the company had yet to conduct any training on its software (it had only been published internally), so the manual had not actually been published with the intent of using it in a classroom. There were some templates already developed for the manual, and I was able to use those—along with the original documentation—as a foundation for the manual.

Bluespring Software’s intent for the manual was that it would accompany a 10-day training course designed to fully educate the existing customers’ AMC configurators on the features and functions of AMC, including configuration management, workflow design, and workflow tasks. Participants used the manual throughout the AMC course to follow along with the instructor, take notes, and perform practice exercises with the actual AMC application. In addition, the manual content was organized in such a way that it could be used as a reference guide after participants had finished the course and begun using the application in the workplace.
The overall objectives for the training manual were:

- to educate configurators about AMC, from installation through advanced topics.
- to help configurators understand each component, or function, of AMC.
- to produce configurators who are familiar and comfortable with AMC and its functions.
- to teach participants the appropriate use and configuration of each workflow task.
- to provide an easy reference for future AMC work.

The AMC training manual consisted of 20 modules that discussed, in appropriate detail, a host of topics from installation procedures to workflow design. Each module contained detailed written instructions, Microsoft PowerPoint slides used during the topic’s presentation (so that participants could follow along during training), and in-class exercises and demonstrations. Each module concluded with a hands-on lab, in which participants created and developed their own project related to the AMC training course.

**Training Manual Design**

Because the AMC Training Manual was designed specifically for classroom training, the original author made deliberate decisions regarding the formatting, layout, and organization of the manual to aid the instructor in presenting the material and to aid the participants in the learning process. As the instructor—as well as the developer of the manual—I had specific goals that I wanted to achieve, and I liked the foundation that the previous author had laid. I chose to continue incorporating her design, which is discussed in more detail below. I liked the fact that I had full creative and content control of the manual, as well as the classroom training format, because it allowed me to integrate the “big picture” of how I wanted the overall training process to evolve and to eventually be executed.

Because the training manual had been written by many authors (originally software developers and then a technical writer), one of my primary objectives when first evaluating it was to determine whether different voices and styles were apparent to the reader. I was concerned that because several people with different writing styles and perspectives had been involved in the development of the training documentation, coursework, and presentations, the integration of all
aspects and materials encompassed in the manual may have been more complicated and less fluent. Often, when several different people contribute to a document (or documents) their different writing styles and techniques can cause confusion and impede learning, rather than facilitating the instructor’s efforts. That is not to say that different perspectives are not important; to ensure that the manual met the needs of a variety of learning styles, I included individuals with different degrees of technological skills, computer knowledge, and career levels in my usability testing for the manual (discussed further in the Usability Testing section of this chapter).

Because I was developing new material and updating much of the existing material for the manual, I wanted to ensure that a consistent style of formatting and a consistent voice, or tone, was used throughout the manual. Readability is critical in technically complex documentation—especially leaning aids—and consistency in style and voice contribute substantially to a reader’s ability to understand the material.

The training manual encompassed three major components in each of its 20 modules: every module contained written descriptions and instructions for the technical functions it covered, MS PowerPoint slides used in the instructor’s presentations so that participants could follow along and take notes, and lab exercises that enabled participants to practice the skills they were learning and more fully engage in the training. In addition, this manual was organized into modules that addressed each of the application’s primary functions so that it could be used as a reference guide after the training course when participants were using the application as part of their jobs. The purpose for the three major components in this manual was primarily to support different learning styles by integrating three methods of instruction. The following sections discuss the design and purpose behind each of the three components in the 20 training modules, as well as prerequisites for the course.

**Requirements for the Class**

The audience for the AMC training course was employees from client corporations who had purchased the software. They would be performing the role of the “AMC Configurator,” the primary user of the AMC software. Before participating in the AMC training course, participants were required to meet several prerequisites to ensure that they had an existing skill set that could serve as a foundation for the new skills that they would be building. In addition, computers that
participants would be operating would also need to meet certain requirements. If the client corporation that was receiving the training could not meet the computer requirements, Bluespring Software made arrangements to conduct the training course at another site. Student and computer requirements were listed in the training manual, but clients were also given this information before Bluespring Software arrived to conduct the training session to ensure that the necessary prerequisites had been obtained.

**Student Skill Set Requirements**
Before they could participate in training, participants were required to possess the following skills:
- Practical understanding of Windows Operating System fundamentals (such booting up the computer, opening applications, transferring files from one place to another, creating folders and saving files, etc.)
- Familiarity with common Microsoft Office Applications

**Computer Requirements**
The following computer hardware requirements were necessary for the participants to participate in the training course:
- 256 RAM
- Pentium II 450MHz CPU
- 1GB Available Hard Drive Space
- TCP/IP Network Connectivity
- 800x600x256 Video

The following computer software requirements were necessary for the participants to enroll in the training course:
- Windows NT 4.0 Workstation SP5 **OR** Windows 2000 Professional SP2 **OR** Windows XP Professional
- Internet Explorer 5.5 SP1 with the Microsoft JVM
Using the training manual, participants verified in class that their computers met the system requirements before beginning the installation process, which was detailed in Module 1: Installation.

**Modules**

As stated previously, the training manual contained 20 modules, with 20 accompanying labs. The modules contained two components: presentation slides and text containing topic descriptions and instructions. Each module in the training manual was centered on a specific component of the AMC and was designed to be a complementary companion to the trainer’s presentation. The MS PowerPoint presentation slides used in the trainer’s presentation allowed participants to follow along in the module as the trainer discussed the material. The excerpt shown in Figure 3-1 below is a page from the training manual; it shows how the PowerPoint slides and text are integrated together.
The MetaModel is at the core of the “configuration without code” architecture of Priority CS. It provides a flexible mechanism for modeling a wide range of business entities. The key to effective configuration is through a thorough understanding of the components of the MetaModel and its properties.

The AMC application is used to manage entity definitions, which are stored within the Priority CS database. The definition not only controls the type, format, and relationships of the data that can be input to an entity, but—to a great degree—it also controls the User Interface (UI) appearance and presentation of that entity to the end user.

**Figure 3-1: Excerpt from AMC Training Manual**

**MS PowerPoint Slides**

The MS PowerPoint slides included in the AMC training manual were the exact slides that were shown in the instructor’s presentation when conducting the class. The purpose for showing the slides in the manual was to allow the participants to more easily follow along with the instructor during the lecture portion of the class and to highlight the major focus points for that particular topic. The focus points were included in the MS PowerPoint slides in the form of bullets. In some instances, the slides also contained graphics displaying elements in the software.
Module Organization
Each of the 20 modules discussed a single topic and contained the following sections:

- Introduction to the topic
- Objectives of that module
- Discussion of the topic
- Introduction to its accompanying lab

Labs
The purpose of the training labs was to enable participants to gain hands-on experience by incorporating the following elements:

- Each module was accompanied by a lab, which gave participants step-by-step practice with the concepts discussed in the module, including using figures and instructions.
- Labs were broken into exercises for ease of use and comprehension.
- Lab assistants would be in the training room to answer any questions participants might have.

Appendix B contains Module 4: Workflow Fundamentals, along with its accompanying lab.

Skill Set Gained by Participants
At the end of training, participants would be able to meet the following objectives:

- Design workflow processes using Priority CS® AMC.
- Understand how to use each component of AMC, from installation through advanced functions.
- Understand the appropriate use and configuration of each workflow task.
- Provide an easy reference for future AMC work.

Resources Included
The following resources were included in the AMC training package delivered to participants at the beginning of each course:

- Training Manual: Each training manual was divided into sections by topic using tabs for easy reference. Training manuals also included space at the end of each module for taking notes during the trainer’s presentation, which participants could refer back to as needed.
• **Appendices:** Three appendices were located at the end of the training manual for participants’ reference in the future. These appendices are designed to act as helpful references as participants configured in AMC.

• **Training CD:** Each training manual came equipped with a CD that contained all of the slides, modules, and labs held in the manuals. The CD allowed participants to access their training manuals.

## Usability Testing

An important part of the completion of the AMC training manual was usability testing. In *Writing Software Documentation: A Task-Orientation Approach*, Thomas T. Barker describes documentation usability testing as conducting “procedures for gaining empirical data about the usability of documentation products.” Because our company didn’t have funding set aside for this project to allow for the time and resources it would take to conduct usability testing on the scale mentioned by Barker, our company had its own method of conducting usability testing that balanced cost-friendliness and the need for objective review and testing of the documentation. When I arrived at Bluespring Software, a usability testing process had already been put in place for previous training materials. Before I arrived at the company, Bluespring Software’s usability testing methods already employed many of the MTSC methods, as Kayte had developed the initial testing processes. While I made some slight modifications, I continued to use her processes as they had been successful in the past and were consistent with my training.

### Usability Tester Selection

According to usability expert Jakob Nielsen (1993), “almost all interfaces need to be tested with novice users, and many systems should also be tested with expert users. Typically, these two groups should be evaluated in separate tests with some of the same and some different test tasks” (p. 177). For the AMC training manual usability tests, I selected employees who met all of the pre-requisites, mentioned below, but whose knowledge of the application spanned a wide range. Obviously, because all of the individuals we selected for the usability test were people who worked at our company, every tester was familiar with the product. However, the participants had never used the product, or at least the portions of the product that they were asked to test. For example, several of the selected users were from the Marketing and Operations departments,
who had little to no experience with using the AMC product. When evaluating whether a person would be suitable as a tester for our usability tests, I evaluated them using the following prerequisites, which I felt would be representative of the training manual’s primary audience:

- Highly proficient in fundamental computer skills and terminology
- Familiar with working in an IT environment
- Minimal exposure to the product

Because anyone using the AMC would be fulfilling a technical, computer-based role at his or her company, the target audience needed to have solid basic computer skills, as well as experience working with various technical tools (hardware, software, programs, etc.) commonly found in an IT environment. If these fundamental skills were not already established, the AMC would be much too advanced for the users. In addition, basic computer skills were not addressed in the AMC Training Manual and needed to be acquired before completing training for the AMC. Finally, I was looking for testers who had not had much exposure yet to the product, as I was trying to confirm whether the AMC training manual was effective for “new” users who had never worked with the product.

Once the usability testers had been selected, I asked them to participate and reviewed Bluespring Software’s Usability Testing Solution with them.

**Bluespring Software’s Usability Testing Solution**

Our usability testing process worked like this: three to four Bluespring Software employees who were not directly working on the AMC software were asked to read the selected module and complete the lab without any additional instructions from myself or other employees. I asked users to record any mistakes or issues they found in the documentation, including, but not limited to, grammatical mistakes, inaccurate or unclear instructions, poor readability, missing information, inability of the instructions to produce the desired result, and effectiveness of screen shots and other figures. I also asked them to note how long it took to read the module and complete the lab for the selected topic. This served two purposes: I was able to gauge how long it took a new user to complete the module and lab, which would help me when planning the
training schedule, and I was able evaluate whether the module and lab were too lengthy for one topic.

I conducted usability testing as I completed each module. Because I was asking Bluespring Software employees to conduct the testing, I knew it would be difficult to find time in their schedules to conduct a formal usability test consisting of all 20 modules. Plus, conducting usability testing as I completed each module and lab allowed me to revise them as they were completed, which sometimes affected subsequent modules and labs (since each topic built on previous modules and labs). Participants only needed to carve out a couple hours a week to conduct the testing. Once the selected participants had completed their testing, I reviewed the areas in the documentation where they had difficulty, and made appropriate revisions. The revised module was then tested again. Each module went through several rounds of revisions, and was tested by several individuals at each round.

In actuality, our usability testing served to test both the documentation and the software. While the software developers were responsible for ensuring their work was complete and accurate, they, too, did not have formal testing. Therefore, users who conducted usability testing on our training modules reported all problems and issues that they encountered. The documentation team was responsible for organizing and conducting user testing because, although it was useful for both developers and technical writers, it was the technical writers who insisted on formalized testing. Since I became solely responsible for the training documentation, I was accountable for organizing usability testing for my project. From a documentation standpoint, I was looking to ensure that the training modules were complete, accurate, understandable, and easy-to-follow. The manuals had several purposes: I wanted them to reinforce training during the course, promote hands-on learning, and act as reference guides when participants were back on the job. I wanted the usability testing to reveal whether the current draft of the manual met these objectives, or whether revisions were necessary in order increase the effectiveness of the manual. The following section discusses results of the usability testing.
Evaluation of Usability Testing Results and Revisions

Once usability testers had completed each module, they gave them back to me along with their comments regarding the content of the module, the presentation slides, and the instructions in the module. I wanted to know whether the modules were informative enough to allow educators to test about the topic so that they could complete the labs without difficulty. As mentioned above, the users were not supervised while testing the manual, but could request assistance if needed. Again, the objectives of the manual were:

- to educate configurators about AMC, from installation through advanced topics.
- to help configurators understand each component, or function, of AMC.
- to produce configurators who are familiar and comfortable with AMC and its functions.
- to teach participants the appropriate use and configuration of each workflow task.
- to provide an easy reference for future AMC work.

In order to determine whether the manual was meeting these objectives, I asked the testers to write down all of their feedback regarding the modules, especially noting any difficulties they experienced when trying to understand topics or complete labs. Although testers often completed the labs over a period of several weeks (due to workload and other priorities), I asked them to complete the modules in a chronological order so that they would get an experience similar to that of a user attending the classroom training. Additionally, the labs built upon each other, and I wanted to test their cohesiveness. I asked primarily open-ended questions to determine whether topics were easy to understand and whether that newfound knowledge transferred into being able to complete labs without difficulty. Testers then returned their comments to me. If I noticed any areas that were especially difficult for a tester, I asked them more detailed questions to try to pinpoint the exact issue so that I could make modifications during revisions.

Overall, the results of the usability testing were very positive. Testers liked the format of the modules and labs, and they felt that for the most part, the manual was easy to follow and understand. They understood how the components of the AMC worked together, and how to configure workflows for different purposes. However, usability testing did reveal several areas of the manual that needed to be revised. There were some instances in which there was too much ambiguity or where directions were confusing. For example, in one of the labs, I forgot to tell users to hit the Enter key on the keyboard—a step that I thought was obvious. However, several
users had problems with the lab, and upon review, I determined that that missing step was causing their errors. In another lab, a simple spelling error caused errors: I misspelled a file name, which resulted in almost all of the testers being unable to find the correct file for their lab. Also, there were a few instances in which there was a “glitch” in the software that caused testers to experience errors.

Once I reviewed the results, I first needed to determine whether the problems encountered by testers resulted from inaccurate or deficient documentation, or whether the software had a bug or other system issue that needed to be fixed. Once I had determined whether the problem was related to documentation or software usability, I then enlisted the appropriate parties to resolve the issue. For example, if ambiguous instructions caused the user to make mistakes in the lab, I revised the instructions for clarity and re-tested the lab to ensure accuracy. Or, if the application was not working as expected due to a software glitch, I asked the responsible programmer to examine the issue and revise it accordingly. The majority of the errors encountered were in the labs, as mistakes were much more apparent there since testers were actively trying to complete a task. When I found an error in a lab, I always made sure to check the corresponding module to ensure that any inconsistencies were correct there, as well.

Conclusion
To summarize, the AMC training manual project allowed me to demonstrate to Bluespring Software that the instructional design, technical writing, and usability testing skills that I had developed throughout the MTSC program were enough to allow me to take on a very large and ambitious project as an intern. Because I had previously learned how to work with technical SMEs and create documentation that was easily understood by a less technical audience, I was able to use my skills, along with the knowledge that I had gained while working at Bluespring Software, to produce a manual that would provide ongoing value for the company (particularly the training department).
Chapter 4: Analysis of Methodologies Implemented: How MTSC Shaped My Internship

Throughout my internship, I structured my work methodologies around the strategies I had studied in the MTSC (and BATSC) programs. Specifically, I employed Paul V. Anderson’s (1995) Problem-Solving Model for Technical Communications in order to define projects and design effective solutions (especially when developing instructional design documents and user testing guidelines). I used this strategy because I had applied it in several projects that I completed during my MTSC coursework, and I knew from experience that it was effective. In addition, the Problem-Solving Model is well-documented as providing an effective framework for determining and implementing solutions to eliminate communication problems. (Indeed, the model is documented extensively in many MTSC internship reports!) Paul V. Anderson’s Problem-Solving Model is composed of the following five tasks:

- Define the problem
- Design the solution
- Test the solution
- Implement the solution
- Evaluate the solution

In this chapter, I will explain how I employed the model while developing and implementing the AMC training manual. In addition, I will reflect upon how I feel my MTSC training influenced my work on the manual, and in a broader sense, my participation at BlueSpring Software.

Defining the Problem

Bluespring Software’s original AMC training manual was a somewhat piecemeal document written by several authors that was never fully executed. Although versions of the document were published internally by the company, it was not a comprehensive manual. Its modules did not encompass all functions of the AMC, and much of the information (including screen shots, instructions, and documentation) was out-of-date, as the software had evolved significantly since the last published revision. Those modules that still contained some relevant information had either not undergone usability testing, or needed to be retested due to changing interfaces and software functions. The primary problem was that Bluespring needed a comprehensive and complete training manual (which included modules describing all functions available in the
software, as well as instructions and practice exercises) to be used during training courses and / or as a reference guide after training had been completed.

A secondary problem was that the SMEs had very limited time to teach me how to operate the software, and I needed to become an expert in order to develop the training manual. Essentially, I was starting out with a knowledge base similar to that of my target audience, but without anyone to train me! In some ways, this was very beneficial, as I think I had a better gauge for the kinds of questions that might arise and how to address them than I would have if I had started out as an expert. In our MTSC courses, we often had become experts in software we had little experience using. For example, I remember spending hours learning Adobe® RoboHelp®, an online help and eLearning system so that I could create online instructions for a website. I also had to learn how to use OneTouch® Blood Glucose Meter so that I could create an instructional manual designed to teach new diabetics how to monitor their blood glucose levels. Studying a variety of software programs, and becoming proficient in them very quickly, enabled me to get up-to-speed on the AMC application quickly because I had experience learning a wide variety of applications throughout my years in the BATSC and MTSC programs.

By defining the problems with the original AMC training manual and contrasting them with Bluespring’s real needs (which were not being met by the current version), I was able to identify which parts of the manual needed to be adjusted and which parts of the existing manual worked well. One of the things I learned from the MTSC program that has benefited me in all of my work experiences is the ability to decipher a client’s (or in this case, a company’s) true needs.

**Designing the Solution**

When designing the solution to my primary problem—developing a new edition of the AMC training manual—I started by first addressing the secondary problem—learning how to use the software and becoming proficient in all of its features. As stated above, Bluespring Software’s SMEs (technical architects, DBAs, software developers, etc.) already had busy schedules with limited time to teach me the “ins and outs” of the application. I decided I would set up hour-long sessions with the SMEs to discuss various topics when they had availability (this often ended up being during their lunch hours). I also chose to use self-study to supplement my own training in
AMC because I knew gaining information on my own when and where possible would lessen my learning curve and expedite the learning process.

As such, I started by scouring the original training manual to glean as much information as possible from that document. I downloaded the software onto my laptop, and stumbled through the original training manual, identifying gaps of information and new updates to the software features, and trying to understand the workflow of the product (software) so that I could construct the workflow of my training manual similarly. As stated in the above section, one of my secondary problems was the limited availability of the SMEs’ time to teach me how to use the AMC applications. Luckily, the MTSC program gave me the tools needed to bravely (because software can be scary) tackle the task of learning much of the software features on my own.

I decided that, where possible, I would construct the manual with a chronological design; the software features and functions often built upon each other, with some of the more advanced features requiring the knowledge and ability to operate some of the more basic features. My intention was to design an AMC training manual that integrated the relevant information contained in the previous manual, while incorporating content that addressed changes to the software, with new modules that incorporated the functions that were either never addressed in the previous manual or that were new to the software’s features. In addition, it was important where possible, for both business and relationship reasons, not to re-do or undo the effort that was previously put into developing the manual. Finally, I intended to keep a similar document design and layout, as user testing had shown it was functional, easy-to-use, and well-liked.

**Testing the Solution**

Much of the testing of the solution was conducted during usability testing, which was previously described in Chapter 3. As stated there, the usability testing required that Bluespring Software employees who were not familiar with the AMC software read a module and complete the corresponding lab. Once they had completed their labs, employees were asked to provide feedback regarding the modules and any difficulties or inaccuracies they encountered.
I evaluated each of the problems encountered and re-worked the material (e.g., documentation or PowerPoint slides) to resolve identified issues. For more complex issues, I often engaged the user (or users) who identified the problem to help me understand how to better frame the documentation to meet their (and future users’) needs. For example, one of the exercises at the end of a module required users to name and save a MS Excel file that they had created. However, the exercise didn’t specify that the file needed to be saved in “.xlm” format rather than “.xls” format in order to function appropriately within the application. Because the default when saving the file was the “.xls” format, most users did not complete the exercise correctly due to faulty instructions. Then, when users attempted to access their file in the next exercise, the program was unable to perform the request. Because this issue spanned multiple exercises (it was created incorrectly in the first exercise of the module, but users weren’t instructed to access the file using the application until several exercises later), it caused problems across multiple modules and made the identification of the true issue difficult. Such trials enabled me to see issues from different perspectives, as well as explain and teach from different perspectives, which I feel genuinely made me a better author and instructor. When writing instructions, it is imperative to ensure they are written clearly and concisely and not leave room for interpretations that could lead to inaccuracies.

**Implementing the Solution**

After designing and testing the AMC Training manual, and after I was satisfied that I had done as much as I was able to ensure the manual was as complete, comprehensive, and accurate as possible, I submitted it to Mr. Trier, my supervisor, for his review. Throughout the development process, I had submitted each of the modules for his “stamp of approval” as I finished them. As the entire project neared completion, I wanted to give him the opportunity to evaluate the project as a whole. Mr. Trier was very satisfied with the final product and asked me to go ahead and “publish” the manual, which consisted of sending it to Kinko’s to be printed on durable, high-quality paper and bound professionally. Kinko’s printed and bound about 50 manuals, which were then ready for the ultimate evaluation: use by actual students in a classroom setting during an AMC training course!
Evaluating the Solution

Unfortunately, I was never able to lead an AMC training course before the end of my internship, and as such, was not present when the AMC manual was put to the ultimate test. I’m confident from the multiple reviews that I had from my supervisor and coworkers that manual was a valuable, easy-to-use learning tool for the course participants. Although I didn’t have the ability to evaluate participants’ reactions to the manuals and obtain feedback personally, I did create written surveys to be distributed to course participants in order to gain valuable feedback which could be used to make any necessary updates to the manual. These surveys requested that participants provide their opinions on the effectiveness, usefulness, and accuracy of the AMC Training manual.

Conclusion

To conclude, the importance and validity of lessons that I had learned through the MTSC program were both validated and reinforced during my internship at Bluespring Software. Specifically, there are three primary lessons that my internship underscored for me:

1. **Technical communicators must be able to adapt their documentation and communication styles to meet users’ needs.** To do so, they must understand their primary audience and how the information in the documentation will be used. They must also understand that documentation may have different purposes for different readers (for example, the AMC manual was used both as a step-by-step guide for new users and as a reference tool for more experienced users) and adapt accordingly by making their text versatile and easy-to-use. While the MTSC program instilled these values in me as a student, my internship at Bluespring Software put them to the test in a “real-world” professional setting!

2. **To be successful, technical communicators must work together as a team toward common goals.** All of the projects completed during my internship required teamwork either as part of the research process (gathering information from SMEs, programmers, etc.), during the documentation’s construction and development (working with other technical communicators to build a product together), or during evaluation and usability testing. It is essential that technical communicators develop skills necessary for interacting both within their own department and across the different departments in a
company. Feedback is an imperative and fundamental tool for technical communicators, and working as a team allows us to constantly provide and receive opinions and guidance regarding our work products, as well as enabling the learning process. Individual perspectives provide valuable insight, but must be contributed in the appropriate context and with the groups’ goals in mind. Technical communicators must be able to collaborate together and with others in the organization to be a successful component of the overall company.

3. **Technical communicators must have an excellent skill set in order to succeed in a professional setting.** As technical communicators, we must be proficient at so many different skills: instructional design, written communication, presentation skills, usability testing, interpretation of technical jargon, graphic design, online documentation, etc. We must have good people skills as well as solid computer- and technical-based skills. And in order to be successful in a professional environment and demonstrate value to our company, we must excel at providing accurate information in a user-friendly format for a variety of audiences. We employ a very unique skill set, and when we use those skills proficiently and expertly, we can make a huge difference to the quality of services that our organizations provide. I observed this during a multitude of client-based projects completed during my MTSC coursework, as well as during my internship. The work that technical communicators perform is important, and if we properly use the skills we have been taught through the MTSC program, and through our professional experiences, our employers will value and reward us for our contributions.

Finally, I’d like to emphasize that my internship experience at Bluespring Software was probably somewhat unique in that the company’s existing processes and documentation standards were quite compatible with my MTSC training. Much of that compatibility was in large part due to the fact that Kayte—who also had a BATSC and MTSC background—had “paved the way” for the technical writing at Bluespring Software. She had educated the company on the multitude of skills that technical communicators bring to their roles, and she had also established credibility for profession by performing well and proving that technical communicators could be a strong asset. Thanks in large part to Kayte, Bluespring Software offered an easy environment in which I could transition from school to full-time work.
Having now left Bluespring to fill a technical writing position at another—much larger—
company, I now appreciate even more Bluespring Software’s dedication to the technical writing
profession. Where another company might have chosen to skimp on filling what are often seen as
“dispensable” roles that can instead be supplemented by programmers or software developers
when their bandwidth allows for it, Bluespring Software made a point to hire highly educated
technical communicators, even when their resources were quite limited. Many of my current
colleagues are under the impression that “nobody reads documentation anyway” and it is
therefore often treated as an afterthought. Indeed, I would argue that one of the reasons technical
communicators aren’t as valued as they should be is that there is a lot of poorly-written
documentation published in the corporate world; however, it’s likely that much of that
documentation was written by individuals who do not have a technical writing background! It is
vital that we, as technical communicators, promote our profession and establish our value not
only by providing high quality documentation and work products, but also by engaging in
professional organizations, networking with colleagues and other professionals, mentoring other
technical communicators, and forging alliances to improve the perceptions of our industry.

I do believe that many companies are starting to recognize that value of strong technical
communicators, and Bluespring Software is certainly helping with that trend. Bluespring
Software valued and rewarded its technical communicators. We were considered an important
component of the overall company, and usable documentation was viewed as a critical to the
success of software products and to the company as a whole. Because Bluespring Software
benefited strongly from the contributions of its technical communicators, it recognized and
rewarded its employees (including myself) who were in the profession. For me, Bluespring
Software was the perfect place to start my career, and the lessons I learned there continue to help
me succeed and persevere in an environment that is often less ideal. I am privileged and grateful
to have had such a rewarding internship experience, which I believe will continue to benefit me
through the duration of my career.
References

