Journal of
Information
Systems
Education

Volume 29
Issue 3
Summer 2018

Moving Forward by Looking Backward: Embracing Pedagogical Principles to Develop an Innovative MSIS Program

Vishal Shah, Anil Kumar, and Karl Smart

Recommended Citation: Shah, V., Kumar, A., & Smart, K. (2018). Moving Forward by Looking Backward: Embracing Pedagogical Principles to Develop an Innovative MSIS Program. *Journal of Information Systems Education*, 29(3), pp. 139-156.

Article Link: http://jise.org/Volume29/n3/JISEv29n3p139.html

Initial Submission: 27 July 2017
Accepted: 5 December 2017
Abstract Posted Online: 13 June 2018
Published: 18 September 2018

ISSN: 2574-3872 (Online) 1055-3096 (Print)

Moving Forward by Looking Backward: Embracing Pedagogical Principles to Develop an Innovative MSIS Program

Vishal Shah
Anil Kumar
Karl Smart
Business Information Systems
Central Michigan University
Mount Pleasant, MI 48859, USA
shah3v@cmich.edu, kumar1a@cmich.edu, smart1kl@cmich.edu

ABSTRACT

Program design is a challenging task that requires significant effort and resources. When a proposed program is being designed for both face-to-face and online delivery, the challenges are compounded. If done right, this task offers an opportunity to leverage pedagogical theory and principles in designing a curriculum for the program. Our research shares findings from a program development effort at a Midwestern university using the Backward Design approach. The Backward Design approach entails working in reverse and involves identification of objectives, creating assessments, and creating learning activities. This approach was used to design a Master's of Science program in Information Systems (MSIS). Alignment of objectives, faculty involvement, mutual accountability, and developing a learning Global Positioning Systems (GPS) for students emerged as key lessons that can be used by other institutions as they undergo efforts to develop or revise curriculum. Further, using Backward Design helped to integrate Assurance of Learning (AOL) processes recommended by AACSB into the curriculum design.

Keywords: Backward design, Program assessment & design, Curriculum design & development, Assurance of learning

1. INTRODUCTION

Program design is a challenging task involving significant effort and resources (Kumar, Shah, and Smart, 2017; Luke, Woods, and Weir, 2013; Winch, 2013). Information Systems (IS), the discipline being addressed in this research, is both interdisciplinary and dynamic as technologies change on a constant basis. Due to the dynamic and interdisciplinary nature of the IS discipline, graduates need to develop an array of skills in diverse areas including technical, communication, and teamwork skills, as well as domain knowledge specific to business (Aasheim, Shropshire, and Kadlec, 2012; Havelka and Merhout, 2009). Prior research in IS has highlighted the interdisciplinary nature of the field, which adds complexity to the process of program curriculum design (Aasheim et al., 2012; Ducrot, Miller, and Goodman, 2008; Havelka and Merhout, 2009). Although extant research has addressed program curriculum design in IS, the focus has been on undergraduate education (Abraham, 2006; Bell, Mills, and Fadel, 2013; Ducrot, Miller, and Goodman, 2008). Limited research on graduate-level IS programs, the changing demographics of students, the learning needs of graduate students, and the dynamic nature of the IS field itself provide an opportunity to advance research in IS graduate program design. Furthermore, most of the research in IS program curriculum design has focused on face-to-face programs. Recent advances in technology and the economy have led to the rise of program delivery in the online channel (Allen and Seaman, 2015). A growing number of working professionals are seeking to enroll in online degrees or certificate programs (Allen and Seaman, 2015; US News, 2016). This trend is expected to grow in a dynamic field such as IS as re-tooling and continuing education lead to significant earning and growth potential (Chong, He, and Wu, 2012; He and Yen, 2014).

Designing a program for multiple delivery models can be challenging if not handled appropriately. For example, in an online setting, the disruptive nature of the medium adds complications (Swan et al., 2014; Yamagata-Lynch, Cowan, and Luetkehans, 2015). A successful program is not "simply about asking faculty to teach existing courses online and go about business as they always did" (Yamagata-Lynch, Cowan, and Luetkehans, 2015, p. 18). In this research, we consider the online dissemination channel in addition to the face-to-face channel. Research on online program design suggests that an effective online program design is a complex function of a myriad of factors, including such things as program delivery, teaching approaches, the quality of teachers, and institutional support (Wiesenberg and Stacey, 2005).

Despite the increase in online programs, their quality has been a concern (Abdous and Yoshimura, 2010; Yang, 2010). An Association to Advance Collegiate Schools of Business (AACSB) report concluded that online learning "requires careful attention to learning design, effective faculty training, organizational commitment to adequate program support, selection of appropriate delivery technology, and a focus on student learning outcomes" (AACSB, 2007, p. 15). If an educational institution seeks to design a program that applies to both face-to-face and online dissemination channels, the process of program design is likely to become more intractable.

Relatively recent research on program design suggests using a collaborative approach based on a well-established theory or pedagogical model (Swan et al., 2014). Effective program design emphasizes the collaborative approach and details how courses in the program relate to each other (Swan et al., 2014; Topi et al., 2016). For this reason, a sound and proven program development based on an established model is critical. Likewise, development must take into consideration the various views and needs of stakeholders (students, faculty, institution, and industry at large) to be successful. Of all potential stakeholders, the need for involving all faculty members in the development process is particularly salient as the natural proclivity of faculty members is to work in silos. There is scant previous research that emphasizes the importance of communication among faculty members (Kim et al., 2012). We found no prior research that documents details of communication structures used during the curriculum development process. Going beyond the ad-hoc nature of IS graduate curriculum research, we describe a model-based approach to graduate IS curriculum development at a program-level. For the program development described in this study, we used the Backward Design approach (McTighe and Thomas, 2003; Wiggins and McTighe, 2001) as the foundation of our work. We also build on recent IS graduate curriculum development research (Topi et al., 2014; Topi et al., 2016).

Backward Design is a widely-used curriculum development technique in education that has been shown to enhance student learning (Bybee, 2006; Childre, Sands, and Pope, 2009; McTighe and Thomas, 2003). This article reports on the curriculum development process at the program-level.

In this research, we describe a model-based graduate IS program developed using Backward design approach. We call this program the Master's of Science in Information Systems (MSIS). The integrated graduate program was developed for face-to-face and online delivery at a Midwestern university. By engaging in such a development process, we answer calls made by scholars in IS and education to develop model-based and integrated curriculum (Swan et al., 2014; Topi et al., 2016; Yamagata-Lynch, Cowan, and Luetkehans, 2015).

Specifically, we focus on two main issues:

1. Designing an innovative graduate program in IS aligned with course objectives and with the mission and vision of the program and the college – at the same time, accounting for faculty members' expertise, proclivities, and preferences while providing students with an engaging and meaningful learning experience.
2. Deducing general principles from the curriculum development process and documenting them so that other institutions can benefit from our experience.

This paper is organized as follows. In section 2, we conduct a thorough literature review of IS program curriculum design research. In section 3, we describe Backward Design and its implementation process along with related documentation. Section 4 lists and discusses lessons learned. In section 5 we conclude with recommendations for other institutions and explain how our work can be extended in the future.

2. LITERATURE REVIEW

As indicated in the previous section, Information Systems (IS) is a multidisciplinary field (Aasheim et al., 2012; Havelka and Merhout, 2009). IS curriculum has reflected the dynamic nature of the field and has responded to job market needs (Jacobi et al., 2014). Recently, program developers have attempted to streamline the IS curriculum development process beginning with Model Information Systems curricula at the undergraduate and graduate levels (Gorgone et al., 2006; Topi et al., 2010). Prior work on systems analysis and design (Guidry, Stevens, and Totaro, 2011) has helped IS educators in preparing better classes for the focal topic. Additionally, there have been recent advances in research on IS security and ERP in undergraduate curriculum at a program-level (Hepner and Dickson, 2013; Patten and Harris, 2013; Woodward, Imboden, and Martin, 2013). Research in curriculum mapping has been a very useful tool for IS educators (Veltri et al., 2011). Also, there has been recent work in the area of combining IS curriculum with liberal arts education to prepare students for work in a complex world (Pratt, Keys, and Wirkus, 2014). Graduate IS curriculum have often been restricted to narrow fields, such as business analytics (Gupta, Goul, and Dinter, 2015). Although prior work in program development has been immensely useful to IS educators, the focus remains on the undergraduate level and addresses specific sub-fields thereby providing an opportunity for IS researchers to advance graduate IS curriculum research at a program-level.

Continuous changes in the field have led IS educators to develop a meta-structure on which specific IS curriculum can be built. The latest example can be seen in modified MSIS 2016 recommendations (Topi et al., 2016). The work of Topi et al. (2016) is of particular importance to IS educators as it provides an overarching structure to build a widely applicable program not bound by specific sub-fields. At the same time, guidelines are relatively fluid and allow the universities to contextualize the program structure. The corpus of work visualizes graduate IS program development based on competencies in each of the three areas depicted in Figure 1.

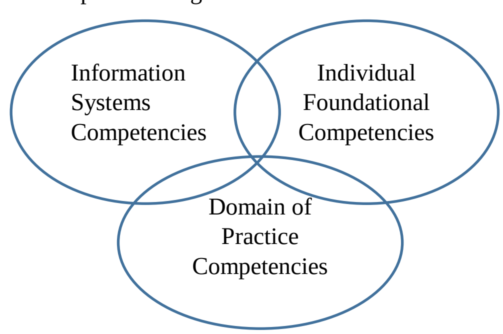


Figure 1. Meta-Structure for IS Graduate Program Development

Our program development differs from Topi et al. (2016) as follows. The competency-based model suggested in their work derives competencies from individual profiles (i.e., business analyst, data analyst, business information manager, etc.), while in our case, we link program-level objectives to competencies.

The difference between Topi et al. (2016) and our approach can be better understood using the following example. Using Topi et al.'s approach, curriculum developers start with professional profiles. To design an IS program, the competencies defined by Topi et al.'s approach will be derived from professional profiles. For example, IS students typically gain employment as a Business Analyst, IT Analyst, Information Manager, Technical Support Specialist, etc. Those designing a graduate program will list a set of competencies that a student would require to fulfill the above-mentioned positions or similar roles. Further, these competencies identified will then be categorized as Foundational, Technical, and Domain (relating to Figure 1). Next, developers create courses wherein they will strive to develop content that helps students master the needed competencies.

Topi et al.'s approach can be very effective in developing courses, and our approach borrows from it. However, instead of deriving competencies from professional profiles, in our approach, we link competencies to college-level learning areas and to program objectives. Learning areas for the college of business, as well as related program objectives, were derived based on the consensus of college leadership, industry partners, faculty members, as well as student representatives. Thus, the manner in which Topi et al.'s approach derives competencies in three areas (Foundational, Technical, and Domain) differs from our methods of arriving at them. It should be noted that our competency categories are not dramatically different from Topi et al.'s work, but we arrived at them by communicating with stakeholders instead of basing them on job profiles.

The primary reason that our results are not different is because these three competency groups advocated by Topi et al. represent the core of Information Systems. In exploring each competency group at a more granular or secondary level, differences may emerge, based on the source (job-profiles vs. stakeholder consensus). However, there is likely to be a considerable overlap as job-profiles represent the need of major stakeholders: employers. Topi et al. (2016) does not restrict the method in which these competencies are mapped to classes. To this end, we employ the Backward Design model.

Having described how our work builds and contrasts with the most recent recommendations on graduate IS program development, we now review additional research on graduate IS program/curriculum. Research on graduate-level program development is scarce (Couger et al., 1995; Gorgone et al., 2006; Jacobi et al., 2014). Much of the current research does not document the communication or the process of development. The conceptual curriculum models are based on a subject-centered approach, student opinions, and alumni surveys, but the models are not applied to the process of course development. In a significant improvement over this approach, recent research (May and Lending, 2015) developed a conceptual model for IS curriculum and applied it to a set of courses, but the development process and general design principles were not readily discernible from the article.

Extant research reports fruitful attempts to generate curriculum based on Bloom's taxonomy (Jacobi et al., 2014; Krathwohl, 2002) and provides a reproducible template to develop courses. The template is highly useful and approaches the problem from the ontological point of view, offering a solution rooted in Business Process Modeling Notation (BPMN) and Unified Modeling Language (UML). Despite being reproducible and useful, the template does not account for various environmental constraints and faculty or department specific idiosyncrasies. As the authors state,

> Currently, the environmental constraints can only be added by means of parameters, and the generator neglects additional practical constraints of interdisciplinary curriculum design. However, a future version of the generator may also handle university specific rules…. (Jacobi et al., 2014, p. 13)

This opens up a possibility for developing and documenting IS curriculum, taking into account institute-related constraints in a non-parameterized way, while ensuring that the developmental process remains modular, so others are able to adopt it.

Much IS curriculum has been based on the key issues derived from prior curriculum research, as well as on triangulation of stakeholders such as alumni, employers, academics institutes, and students (Chiang, Goes, and Stohr, 2012; Lee, Trauth, and Farwell, 1995) – for example, the Skills Framework for the Information Age (SFIA) (von Konsky, Miller, and Jones, 2016). SFIA facilitates interaction among faculty members, thus developing relevant curriculum. The introduction of such a framework is a welcome addition towards developing relevant graduate classes. However, classes developed within such a framework may not lead to the desired curriculum for several reasons. While the content in the course may be in alignment with the course objectives, the faculty member may not clearly appreciate or understand how their course fits at a program-level. Generally, faculty members are inclined to work in isolation, and such a situation reiterates the importance of the communication among faculty members in the curriculum-development process. We found no significant prior research providing detailed documentation of the curriculum development process as it occurred between faculty members.

In addition to providing a curriculum development structure that promotes communication among faculty members, it is important to document these practices so that the lessons learned in IS curriculum development can be generalized and adopted by other institutions. Some prior work (Kim et al., 2012) has provided guidelines for developing IS curriculum, but they were specific to ABET accreditation and not in the business school context.

This study demonstrates our approach to curriculum development at the program-level, in keeping with the institution's mission and values, and follows a process-based philosophy. A large corpus of research on IS graduate curriculum development is ad-hoc and scattered in nature. There is an opportunity to conduct meaningful research in IS graduate curriculum development focused at a program-level and addressing both modes of delivery (face-to-face and online). We build on this research gap to devise a graduate IS curriculum as well as provide documentation and recommendations that may help other academic institutions develop their graduate IS program using a more robust, research-based model. As discussed in the earlier section, we use the framework called Backward Design, and specifically, we build on recent IS graduate curriculum development research (Topi et al., 2014; Topi et al., 2016). The following section describes the Backward Design process.

3. BACKWARD DESIGN

Backward Design represents a paradigm shift in curriculum design (Wiggins and McTighe, 2001). From a design perspective, Backward Design enhances student learning. Backward Design ties course objectives to assessment and corresponding learning activities. This reverse engineering approach provides a clear "roadmap" for designing and organizing course content to achieve the focal course objectives. Designing curriculum based on Backward Design involves three phases as depicted in Figure 2. Although designing curriculum based on desired outcomes is not a recent idea (O'Neil, 2010), IS research in this area has been ad-hoc and scattered. We apply the well-established Backward Design curriculum design model to the IS curriculum process as well integrate it with recent IS curriculum development research (Topi et al., 2016). This research answers "why" this work is important and "how" we implemented it (Yin, 1994). During the application of Backward Design, in addition to the elements recommended by the model (objectives, assessment, and activities), we paid particular attention to the process of consensus formation and accountability among curriculum development team members. We meticulously documented deliverables relevant to each stage of Backward Design.

3.1 Facilitating Integrated Development

To facilitate faculty communication and mapping of dependencies, we created three cohorts (Foundational, IS, and Domain). Each cohort had a cohort coordinator. The cohort coordinator met with the program director on a regular basis to discuss desired outcomes, evidence of learning (i.e., assessments), and learning activities. The cohort coordinator conveyed the agenda of the respective group to the program director. The program director ensured that there was synchronicity between various groups. Reporting structures can be seen in Figure 3. Each cohort consisted of a set of courses designed to impart competencies in the respective area (Foundational, Information Systems, and Domain of Practice) advocated by recent research in IS graduate curriculum (Topi et al., 2016).

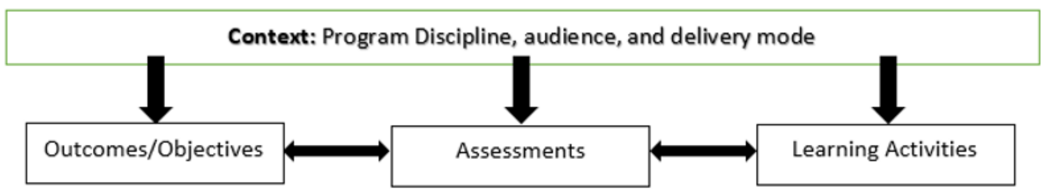


Figure 2. Backward Design Model

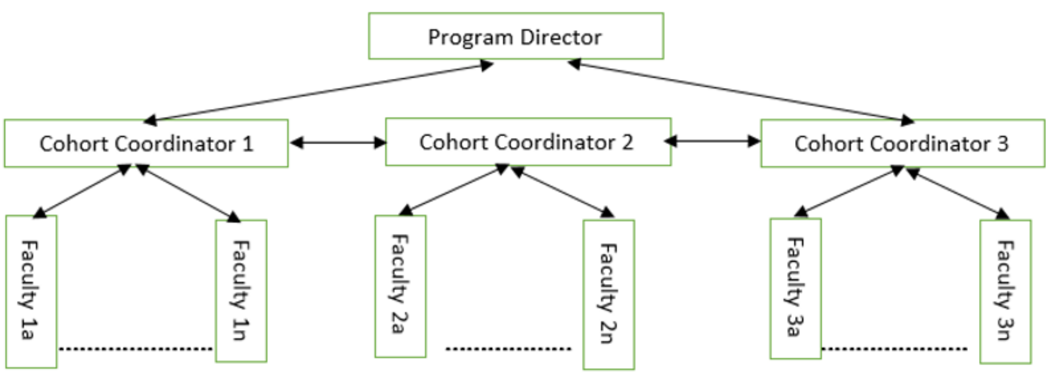


Figure 3. Facilitating Mechanism

As we will explain in the following section, we kept a detailed log of activities. This documentation was shared with the entire program curriculum design team (i.e., all faculty members) to ensure that they were aware of the change and progress. This activity ensured that all faculty members were included in the process, which is of paramount importance to the overall success of the MSIS program. Careful mapping of classes and interdependencies between them led to the development of a coherent program as well as assuring mutual learning while accounting for individual faculty member preferences. Figure 4 depicts a granular (course level) version of our integrated curriculum but does not indicate feedback loops for maintaining interpretive clarity. In the following paragraphs, we illustrate how the most recent curriculum development (Topi et al., 2016) relates to program development in our context and the manner in which the development process facilitated integrated program development.

We achieved this by combining various courses into cohorts – a foundational cohort, an IS cohort, and a domain cohort – corresponding to the three competency areas suggested by recent IS research (Topi et al., 2016). Tables 1, 2 and 3 show the mapping of the courses in our program and competency domains. It should be noted that each class likely maps into more than one competency category. However, one main category is pronounced in a particular class and is correspondingly marked with P (indicating primary). Other categories become secondary and are marked with an S (indicating secondary). Some courses may have more than one primary competency. In such instances, the primary competency that belongs to the corresponding cohort (indicated by a stand-alone P) should be most relevant from the instructional standpoint.

As an instructor of a course, the faculty member aims to develop the primary competency related to the focal group under consideration. The other primary competency (belonging to other focal areas) should be understood as utilized in that class and not developed during the class. It is marked as P (utilized), indicating it was derived from earlier classes. Whenever a competency is utilized, it reinforces itself from a cognitive standpoint. The same explanation holds true at the

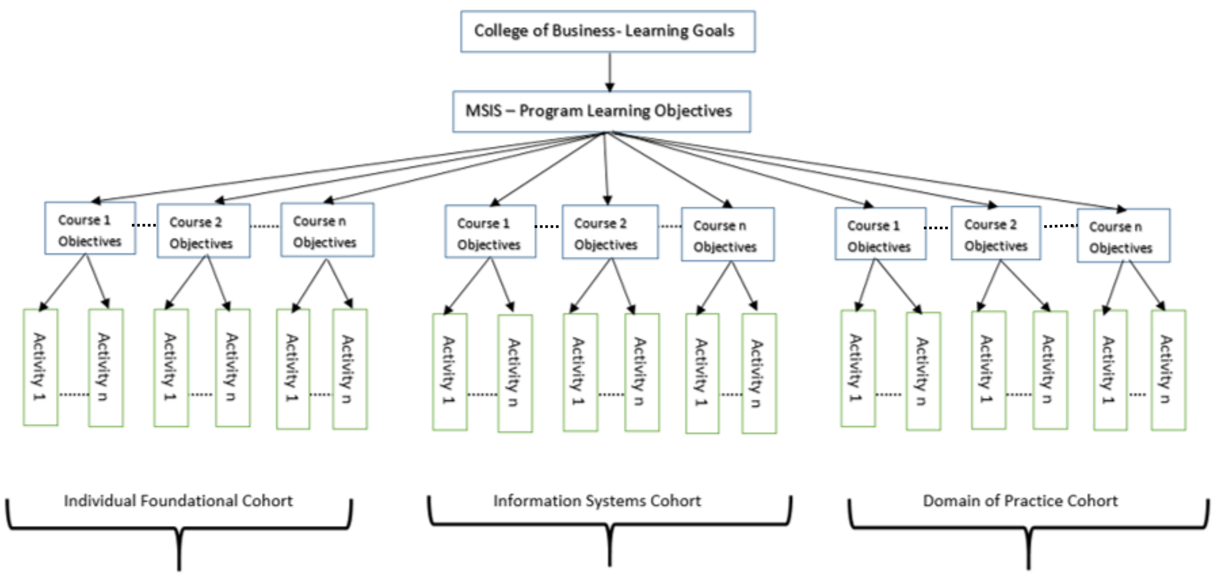


Figure 4. Detailed Program Structure

Course	Various Possible Competencies: Individual Foundational	Information Systems	Domain of Practice	Foundational Focal Competency Cohort
PROJ-MGT Project Management	P	S	S	**Individual Foundational** These three classes form a group which primarily addresses Individual Foundational Competencies. Together these classes familiarize students with critical thinking, collaboration/teamwork, intercultural communication, ethics, problem-solving, and written/verbal communication. These competencies are not technical but rather a pre-requisite to succeed in today's workplace, regardless of the field of study.
IS-INTRO Introduction to Information Systems	P	S	S	
IS-COMM Business Communications	P		S	

Table 1. Connecting Various Competencies and Courses in Foundational Core

Course	Various Possible Competencies			IS Focal Competency Cohort
	Individual Foundational	Information Systems	Domain of Practice	
APP-PROG Applied Programming	S (utilized)	P	S (utilized)	**Information Systems** The four classes in this cohort together form "IS core" group. The rationale behind integrating these four classes is to provide students with the majority of IS competencies referred in AIS Task Force's recent graduate IS curriculum.
SYS-DESG System Analysis and Design	S (utilized)	P	S (utilized)	
DB-SYS Database Systems	S (utilized)	P	S (utilized)	
CAP-PROJ Capstone Class	P (utilized)	P	S (utilized)	

Table 2. Connecting Various Competencies and Courses in the IS Core

secondary level. Note that the students are advised to take "foundational core" classes before taking "IS core" classes; they then proceed to elective classes. The MSIS degree consists of twelve, 3-credit courses.

The concept of the foundational core can be shown through an example from the Project Management class. In the project management class (PROJ-MGT), a class project requires a formation of a team. This team may consist of students of various educational and national backgrounds, as the program is open to international students. PROJ-MGT focuses on the foundations of project management. Students are expected to work together, negotiate workload, develop an understanding of project management techniques used to assess projects (e.g., PERT and Gantt charts), and present the outcome of the project as a team. Going through this process requires that students develop competencies in the foundational areas. The other two classes also are developed in the same vein, with IS-INTRO introducing students to basic IS concepts relevant to organizations. IS-COMM focuses on investigating the role of communication in a professional context and developing core abilities such as audience analysis, writing, presentations, interpersonal communication, and intercultural communication. Table 2 shows the connection of various competencies in the IS core classes.

The integration of four courses (APP-PROG, SYS-DESG, DB-SYS, and CAP-PROJ) forms the IS core. For example, APP-PROG equips students with programming fundamentals and programming for business problems, whereas SYS-DESG focuses on the process of designing and developing information systems. DB-SYS is a database class equipping students with the knowledge needed to design and build a database. The capstone class, CAP-PROJ, familiarizes students with IS management, operations, enterprise architecture, and systems development. IS competencies are built and emphasized among these four classes. Again, as with the foundational core, competencies (highlighted in the foundational cohort) are also utilized and reinforced in each class in this cohort as well, allowing us to build a more integrated program.

The previous two cohorts (foundational and IS) constitute our "core classes." The third cohort focuses on using domain knowledge. In the curriculum, we offer three domains or concentrations: Enterprise Systems, Project Management, and Cybersecurity. Students can choose one of the three concentrations, each consisting of three classes. The Cybersecurity concentration allows students to choose three classes from four options as shown in Table 3. These concentrations were designed with the objective of providing students expertise in specific areas leading to industry certifications. For example, the enterprise system concentration prepares students to take Enterprise Resource Planning (ERP) certification called TERP-10 offered on campus and administered by SAP.

As explained earlier, all students are required to take core courses forming Foundational and IS competencies. Learning in Foundational and IS competencies is assessed using the capstone project. As different students may choose different specializations, we do not include courses in specializations as part of the capstone project. Instead, the courses in the concentration help students prepare for industry certifications and form domain competencies.

Our graduate program is a general IS degree that is designed for candidates with diverse backgrounds to gain foundational IS knowledge and specialize in one of the three concentrations. There is no prior expectation of any IS-related background. The MSIS program follows the suggestion of a minimum 18 months of full-time study and a minimum of 36 credits as illustrated earlier. In the U.S. education system, 36 credits translate to 12 courses of 3 credits each. Topi et al.'s (2016) proposed structure makes it clear that there is no prior expectation of professional experience. This assumption is of particular importance as an MSIS program draws students from a variety of backgrounds. In the same vein, our program is a designed as a "pre-experience" program. Current enrollment data indicates that international students form a significant portion of the face-to-face program whereas domestic students have a higher representation in the online version.

Topi et al. (2016) point out the issues with the hierarchical Knowledge Area–Knowledge Unit–Topic (KA/KU/Topic) structure that forms a Body of Knowledge (BoK). The KA/KU/Topic structure de-emphasizes the experiential elements while overemphasizing cognitive elements. The central problem that Topi et al. (2016) points out is that program development has been very course specific, and not enough information is provided on how individual courses tie to program-level objectives or capabilities:

> The main challenge of this approach is that it typically presents a course-specific view without providing a

Course	Various Possible Competencies			Domain Focal Competency Cohort
	Individual Foundational	Information Systems	Domain of Practice	
Enterprise Systems				
ES-BPI INTRO Introduction to Business Process Integration	S (utilized)	S (utilized)	P	This concertation focuses on using enterprise systems for business. We use SAP for enterprise systems. This concertation deals with business processes in detail; it also educates students in configuration and programming of such enterprise systems.
ES-PROG Programming in Enterprise Systems	S (utilized)	S (utilized)	P	
ES-COFG Configuring Enterprise Systems	S (utilized)	S (utilized)	P	
Project Management				
PROJ-MGT PI Business Process Improvement	P (utilized)	S (utilized)	P	The Project Management concentration further focuses on project management methodologies, including Lean Six-Sigma. It also focuses on staffing and behavioral and qualitative skills in a simulated project to develop core project management competencies. It draws heavily on foundational skills.
PROJ-MGT ADV Application of Project Management Principles	P (utilized)	S (utilized)	P	
PROJ-MGT ADV II Advanced Project Management Methodology	P (utilized)	P (utilized)	P	
Cybersecurity				
CSEC-FUND Network and Systems Security Fundamentals	S (utilized)	P (utilized)	P	The Cybersecurity concentration has one required class (CSEC-FUND), which familiarizes students with effectively and accurately analyzing security risks related to networked systems, including issues of authenticity, confidentiality, integrity, and availability. The other two classes can be chosen from the three options. The Cybersecurity concertation builds heavily on IS core competencies and utilizes foundational competencies as shown. The reason for heavy reliance on IS core competencies is due to the fact that students go through the exercises of finding faults, working with command line interface, and recovery of files and emails.
CSEC-GOV Governance, Risk, and Compliance in Cybersecurity	S (utilized)	P (utilized)	P	
CSEC-FORC Cybercrime Forensics	S (utilized)	P (utilized)	P	
CSEC-CLD Managing Privacy and Security in the Cloud	P (utilized)	P (utilized)	P	

Table 3. Connecting Various Competencies and Courses in the Domain Cohorts

detailed program-level representation of expected graduate capabilities. Some of these curricula, such as IS 2010 dedicate significant attention to the specification of program-level graduate capabilities at a high level of abstraction, but even IS 2010 never maps the course level with the program level to analyze or demonstrate how the courses contribute to the way in which students achieve the program-level objectives. (Topi et al., 2016, p. 7)

We remedy this issue by mapping program-level objectives to course-level objectives and providing how each course objective is assessed. Further, activities in a specific course are also derived using a template based on course-level objectives. Thus, our approach allowed for the development of the courses in an integrated manner that was synchronized at a program-level.

Phase 1	Phase 2	Phase 3
Outcomes: Identify desired results of MSIS program/Program Level Objectives (PLOs)	Assessment: Determine acceptable evidence for MSIS program	Design: Plan learning experiences and instruction in MSIS courses
What is it that we want students to understand, know and be able to do?	How will we know that students know what we want them to know?	What do we need to do in the classroom to prepare students for the assessment?

Table 4. Phases of Backward Design

3.2 Implementing Backward Design

The three phases of Backward Design mentioned earlier are described in more detail in Table 4. Phase 1 specifically dealt with finalizing program-level objectives (PLOs). The PLOs were developed after extensive discussions by faculty in the department using AIS recommended MSIS curriculum core areas (Topi et al., 2014; Topi et al., 2016). Using AIS as the basis for our curriculum development ensured that the graduate program reflected key priorities identified by the AIS task force. These PLOs themselves were mapped to college-wide learning areas and course learning goals. The college-wide learning goals were identified after an iterative process that involved college faculty and external stakeholders. This facilitated the process of alignment of PLOs with college learning goals. In this phase, our objective was to ensure that guidelines suggested by the AIS taskforce were incorporated in our context while ensuring stakeholders' needs were met. Table 5 provides the mapping of program-level learning objectives with college-wide agreed upon learning areas.

The next step was to map program-level objectives to course-specific student learning objectives (SLOs). Table 6 provides the detailed documentation of this process. Each faculty member was responsible for providing the course SLOs. Again, these SLOs considered faculty-specific proclivities but were not derived in a vacuum; rather, they were consensus-based following the coordination structure described in Figure 3. This structure facilitated bidirectional communication both within and between cohorts at a program-level. All faculty members were aware of the ongoing status of the development which facilitated consensus.

Phase 2 of the Backward Design process dealt with establishing acceptable evidence of learning, ensuring that course-specific SLOs were assessed. Integrating assessments with course SLOs and mapping of these course SLOs to PLOs made mapping apparent and manageable at this stage. In this phase, the deliverable for each class in the respective cohorts was a listing of assessment techniques appropriate for the focal class. Addressing assessment as part of the program design initiative helps align with AACSB Assurance of Learning (AOL) standards. AACSB is the premier accrediting body for business programs.

In Phase 3, faculty developed learning activities and instructional strategies for their respective classes. Table 7 provides the template for designing learning activities and corresponding assessments (acceptable evidence) to evaluate it. For example, assessment for a specific class could range from quizzes, a qualitative test, a project or presentation, to any combination of such measures. Activities are designed using the "WHERETO" method. This method ensures that learning activities are thoughtful, engaging, reinforcing, and organized. The "WHERETO" method has been used extensively in the field of education (Daugherty, 2006; Wiggins and McTighe, 2005). Again, as in the previous phase, these decisions were made by faculty consensus following Figure 3 and Figure 4, while making all attempts to account for individual faculty preferences. Each letter in "WHERETO" provides an anchor. Learning activities were designed based on these anchors. For example, the letter "W" focuses on three questions: 1) What are students learning? 2) Why are they proceeding in a specific direction? and 3) What are the ways in which they will be evaluated once they do the activity? This, in addition to other

College of Business Administration (CBA) learning goal areas	MSIS Student Learning Objectives (SLOs)			
	MSIS SLO1: Demonstrate the ability to collaboratively develop information systems to solve business problems.	MSIS SLO2: Demonstrate the ability to provide accurate information to key stakeholders	MSIS SLO3: Demonstrate competency in core business knowledge.	MSIS SLO4: Demonstrate effective business writing and oral communication skills.
CBA#1 Business Knowledge			x	
CBA#2 Business Communications		x		x
CBA#3 Problem Solving	x	x		
CBA#4 Information Technology	x	x		

Table 5. Mapping of CBA Learning Goals with MSIS SLOs

Course	Course SLO	MSIS SLO #1	MSIS SLO #2	MSIS SLO #3	MSIS SLO #4
PROJ-MGT	Plan, execute and monitor a project.	x			
	Explain the basic interactions of initiating, planning, executing and monitoring.	x			
	Solve typical business project issues utilizing learned project management skills.			x	
	Describe project communication and stakeholder management.		x		x
IS-INTRO	Explain the purpose and role of information systems.		x	x	
	Combine application software using spreadsheets, database, presentation and word processing to solve business problems.		x	x	
	Illustrate knowledge of integrated information systems implementation.		x	x	
IS-COMM	Articulate the role and value of communication abilities for information systems professionals.		x		
	Demonstrate effective business writing skills.				x
	Demonstrate effective oral communication skills.				x
APP-PROG	Analyze business problems applying logical reasoning to break it down into its component parts.	x			
	Apply programming knowledge to develop business solutions.	x	x	x	
	Create interfaces that enable.	x	x		
SYS-DESG	Apply the key concepts of systems analysis and design to develop a prototype for a computer-based information system.	x			
	Develop information systems architecture that meets business needs to solve organizational problems.	x			
	Demonstrate effective business writing and oral communication skills.				x
DB-SYS	Develop a data model based on analysis of user requirements.	x			
	Transform a data model into a well-structured relational database.	x			
	Process/Operate a relational database using an appropriate query language.		x		
CAP-PROJ	Apply systems development concepts and methodologies to collaboratively design and develop information systems to meet business needs.	x	x	x	
	Demonstrate effective business writing and oral communication skills.	x			

Table 6. Aligning Core Course Objectives to MSIS Program Objectives

anchors in the template, provided a streamlined approach to the development of learning activities. Here it is relatively easy to see the iterative nature of the program development process. It may be possible that assessment technique(s) that the individual faculty member, cohort coordinator, and program director agreed upon in Phase 2 may turn out to be inadequate as the group cogitates the design of learning activities. In such a case, assessments are revisited so that they are streamlined with the following phase. Thus, the process is iterative. First, the entire group goes through all three phases linearly. After that, the process becomes iterative based on further interactions, preferences of faculty members, and discussions in the light of course SLOs.

The iterative process of consensus building and justification for learning activities and assessment methods provided fluidity and program integration; at the same time, the curriculum development was based on well-defined models and templates. The continued tension between consensus dialectics and structure ensured that courses in each cohort were integrated,

<table>
<tr><td>Course SLO's</td><td rowspan="2">Teaching and Learning Activities

W = How will you help your students to know where they are headed, why they are going there, and what ways they will be evaluated along the way?

H = How will you hook and engage students' interest and enthusiasm through thought-provoking experiences at the beginning of each instructional episode?

E = What experiences will you provide to help students make their understandings real and to equip all learners for success throughout your unit or course?

R = How will you cause students to reflect, revisit, revise, and rethink?

E = How will students express their understandings and engage in meaningful self-evaluation?

T = How will you tailor (differentiate) your instruction to address the unique strengths and needs of every learner?

O = How will you organize learning experiences so that students move from teacher-guided and concrete activities to independent applications that emphasize growing conceptual understandings?</td></tr>
<tr><td>Acceptable Evidence</td></tr>
</table>

Table 7. Learning Activities and Acceptable Evidence Template

and in turn, each cohort logically related to others. Table 8 provides an actual snapshot of how activities were designed in a specific class (DB-SYS) based on the previous template.

Backward Design places assessment at the center of program design. The assessment method was agreed upon by the focal faculty member, the rest of the cohort members, and the program coordinator for a particular course. To ensure coherency of the program, assessment methods were also mapped to a specific course objective, which itself was mapped to MSIS SLO(s). Table 9 provides the detailed documentation.

As can be seen from Table 9, one assessment method can serve multiple course objectives (SLOs). However, not all courses will cover all four MSIS objectives (PLOs). This fact can be seen in the example of IS-INTRO, which emphasizes PLO#2 and PLO#3. However, all courses taken together cover all four MSIS PLOs.

3.3 Emphasizing Core Learning

As a program-level decision, the development team decided all MSIS graduates should be conversant with the knowledge imparted in the Foundational and Information Systems cohorts. To implement this, the core curriculum map (Table 10) was developed. It indicates where a specific PLO was introduced (I), reiterated (R), emphasized (E), and assessed (A).

4. LESSONS LEARNED

The Backward Design approach to curriculum design helped us comprehend and develop the MSIS program curriculum. In this process, we identified and addressed multiple challenges. Other educational institutions trying to develop innovative programs can learn a significant amount from the challenges that we faced. In the following paragraphs, we share and discuss these challenges and takeaways we learned.

4.1 Alignment

Schmidt-Wilk (2011) argues that the process of course design can be viewed as a strategy. We believe that not only course design, but program design, can be viewed as a strategic process. In this process, it is important to make sure that the views of all the stakeholders needed to implement the strategy are aligned. Alignment of stakeholders' views ensures that they reinforce the efforts of one another leading to successful implementation of the strategy, i.e., program goals.

For an academic unit (college or department) developing an academic program, alignment happens at multiple levels. The academic unit needs to make sure that they align their program learning goals with the college vision and mission and at the

Course SLOs Develop a data model based on analysis of user requirements. Transform a data model into a well-structured relational database. Apply SQL to manipulate and retrieve data from a relational database.	**Teaching and Learning Activities** W = How will you help your students to know *where* they are headed, *why* they are going there, and *what ways* they will be evaluated along the way? *The course syllabus contains a week-by-week schedule listing topics to be covered. This schedule will be referenced during each instructional episode to make sure students know where they are headed.* *The course introduction will explain that the course is divided into three parts: relational database theory, SQL, and design/development practicum. The introduction will go on to explain that each part of the course builds upon the last (i.e., the foundation is built on theory; the theory is applied through the use of SQL, and finally theory and SQL are used to design and develop a complete database. This will serve to help students understand why they are going there.* *The course syllabus will explain how problem sets and exams will be used to assess the mastery of theory and SQL.*
Acceptable Evidence (Assessment) Problem set Exam	H = How will you *hook* and engage students' interest and enthusiasm through thought-provoking experiences at the beginning of each instructional episode? *Each instructional episode will begin with a thought question or scenario that will serve as the motivation for the material to be covered.* E = What *experiences* will you provide to help students make their understandings real and to *equip* all learners for success throughout your unit or course? *Students will complete problem sets based on the theory and SQL material that is covered. These problem sets will challenge students to apply the knowledge they have gained to solve real database design/development problems.* R = How will you cause students to *reflect, revisit, revise, and rethink*? *Students will be given the opportunity to submit drafts of design documents for review. Feedback will be provided, and the students will have the chance to revise the documents after reflecting on the feedback.* E = How will students *express* their understandings and engage in meaningful self-*evaluation*? *Students will be given practice problem sets that they can choose to complete. Answers to these problem sets will be provided so the students can evaluate their own performance while identifying any areas of concern.* T = How will you *tailor* (differentiate) your instruction to address the unique strengths and needs of every learner? *Instruction will be consistent for all students. Additional instruction and individualized feedback will be provided as needed to address specific student needs.* O = How will you *organize* learning experiences so that students move from teacher-guided and concrete activities to independent applications that emphasize growing conceptual understandings? *As stated above, the course is divided into three parts. The first two parts cover the theory of relational databases and the use of SQL. Once students have been exposed to this material (through lectures and multiple problem sets), they are required to apply what they have learned through a design/development practicum.*

Table 8. Learning Activity Example from Course DB-SYS – Database Management

same time ensure alignment with the student learning objectives of each course in the program. This alignment should enable individual faculty to see how their course fits into the overall program and how the program is aligned with the vision and mission of the college. Further alignment of PLOs with a college vision and mission helps gain institutional support, whereas alignment of PLOs with individual course SLOs ensures faculty support. Costigan and Brink (2015) suggest that misalignment between program goals and curriculum raises the question of "relevance of an academic program."

When we started the design process, we made sure that all faculty responsible for teaching a course in the program were invited to join a cohort. The course teaching responsibilities were the result of faculty volunteering to teach a course based on their interest rather than an arbitrary assignment by the department. An email soliciting faculty interest was sent and, based on responses, a master list was created for teaching

Course	Course SLO	MSIS SLO#1	MSIS SLO#2	MSIS SLO#3	MSIS SLO#4	Assessment
PROJ-MGT	Scope, plan and execute a basic project.	x				Project Simulation
	Explain the interactions of the Project Management Process Groups and Knowledge Areas.	x	x		x	Weekly quizzes Pre-CAPM exam
	Solve typical business project issues utilizing learned project management skills.			x		Project Simulation Weekly assignments
IS-INTRO	Explain the purpose and role of information systems.		x	x		Reading Summaries Problem sets Exams
	Demonstrate the use of applications including spreadsheet, database, presentation, and word processing to solve business needs.		x	x		In-class activities Problem sets Exams
	Demonstrate knowledge of integrated information systems implementation.		x	x		Reading Summaries Problem sets Exams
IS-COMM	Describe the role and value of communication abilities for information systems professionals.		x			Research report (primary and secondary research)
	Demonstrate effective business writing skills.				x	Short report and Business memo
	Demonstrate effective oral communication skills.				x	Presentation
APP-PROG	Analyze business problems applying logical reasoning to break them down into their component parts.	x				Problem sets
	Apply programming knowledge to develop business solutions.	x	x	x		Project Assignments
SYS-DESG	Apply the key concepts of systems analysis and design to analyze problems and gather requirements for organizational needs.	x		x		Prototype Business requirements specifications Discussions Case study
	Create information systems architecture that meets organizational requirements.	x		x		Business IT architecture Discussions Case study
	Produce and communicate an effective business plan.				x	Business plan
DB-SYS	Develop a data model based on analysis of user requirements.	x				Problem set Exam
	Transform a data model into a well-structured relational database.	x				Problem set Exam
	Apply SQL to manipulate and retrieve data from a relational database.		x	x		Problem set Exam Project
CAP-PROJ	Apply systems development concepts and methodologies to design and develop information systems collaboratively to meet organizational requirements.	x	x	x		Project
	Demonstrate effective business writing and oral communication skills.				x	Project proposal Project plan Project presentation

Table 9. Mapping Course(s) Assessments to Core Course(s) SLOs and MSIS PLOs

CBA Learning Goals	Corresponding MSIS Program Learning Objectives (PLOs)	Information Systems Fundamentals			Information Systems Development			
		PROJ -MGT	IS-INTRO	IS-COMM	APP-PROG	SYS-DESG	DB-SYS	CAP-PROJ
CBA #1. Business Knowledge. Demonstrate competency in core business concepts, knowledge, and chosen discipline needed in a global environment.	MSIS PLO 1. Demonstrate competency in core business knowledge.	R	I		R	R	R	E Assessed
CBA #2. Business Communication Display effective business communication skills.	MSIS PLO 2. Demonstrate effective business writing and oral communication skills.	R	R	I, E Assessed		R		E Assessed
CBA #3. Problem Solving. Demonstrate the ability to use business information and solve business problems.	MSIS PLO 3. Demonstrate the ability to collaboratively develop information systems to solve business problems.	R	I		R	R	R	E Assessed
CBA #4. Information Technology. Apply technology skills to business problems.	MSIS PLO 4. Demonstrate the ability to provide accurate information to key stakeholders.	R	I	R	R		R	E Assessed

Table 10. MSIS Program Curriculum Mapping

assignments. In some cases, multiple faculty members signed up to develop a course. Approval was sought from the faculty to ensure that they agreed to the collaborative development in cases where multiple faculty members signed up for a course. The process ensured that faculty understood what was happening in the process, leading to the development of shared vocabulary and understanding.

Regular meetings of the cohorts provided constant communication, which often resulted in extended discussions. Faculty members were forthcoming in sharing their concerns, and often had disagreements on alignment. Individual faculty responsible for teaching a course helped cohort members understand how they were addressing alignment of their course SLOs to PLOs. Transparency of the process enabled faculty to see value and potential in differing viewpoints. Colleagues can be instrumental in offering honest and valuable feedback when they understand the process. It is important that in efforts such as the one discussed in this paper, colleagues evaluate each other's viewpoints. Despite differences in perspectives, we discovered that it is easier to resolve disagreements when peers are involved. We assert that in our program design efforts, the constant communication and transparency of the process for various stakeholders enabled us to address differences in viewpoints. This facilitated achievement of strategic alignment of PLOs with college vision and mission and course SLOs.

Lesson: Communication, clear value, and transparency enable alignment.

4.2 Faculty Involvement

Earlier we explained that alignment could not happen without faculty involvement. Bringing together a group of academics may prove akin to "herding cats," as academic work requires creative thinking that often leads faculty members to work in isolation. Natural synergies occur when common research interests prompt faculty members to work together. However, a similar level of participation may not be as prevalent in the curriculum development process.

Faculty resistance is a documented phenomenon when it comes to online program development and use of novel program development approaches (Allen and Seaman, 2010; Appana 2008; Watty, McKay, and Ngo 2016). Given this proclivity of faculty members as well as time constraints, it may be challenging to involve them in a collaborative curriculum development effort. Thus, we created a facilitating mechanism that fostered participation and encouraged sharing of opinions. As depicted in Figure 3, we created multiple manageable faculty cohorts mapped to the three underlying domains. Such an approach ensured that faculty members were not overwhelmed with the information that was not of primary importance to their cohort in the initial stages. The facilitating mechanism permitted each cohort to share their work and concerns with the larger group while allowing faculty members to work in the relative isolation of their respective cohorts.

Again, generating faculty buy-in and participation was not a linear progression, and we had to improvise as well as create a nonjudgmental environment where disagreeing opinions could be shared. Involving faculty members who teach in a program in the program-design process ensured that individual faculty members understood the role of their course in the context of the program, as well as motivated them to contribute to the program. The faculty in this process can be compared to the learner, where involvement in decision-making is perceived as a learner-centered approach to learning.

Lesson: Divide work into manageable chunks while maintaining individual faculty ownership.

4.3 Teamwork Leading to Mutual Accountability

Creating shared meaning and vocabulary helped us to develop two critical components that lead to successful program design: 1) Collective Responsibility and 2) Peer Monitoring.

4.3.1 Collective responsibility: Collective responsibility refers to a type of prosocial behavior where individual members of a particular group take responsibility for the other members' behavior because of social association with the group (Hamilton, 1978; Lickel, Schmader, and Hamilton, 2003; Sanders et al., 1996). Collective responsibility can be seen as an informal control mechanism **(**Diefenbach and Sillince, 2011). As discussed earlier, it is difficult to achieve compliance from faculty members given the nature of academic work and differing viewpoints. Enforcing formal and rigid control mechanisms can be counterproductive and lead to faculty resistance, as many faculty members can be averse to the change (Allen and Seaman, 2010; Appana, 2008; Watty, McKay, and Ngo, 2016). Thus, we sought to involve faculty members using shared program vision instead of rigid policy enforcement. Emphasizing and reiterating program vision, willingness to listen to faculty concerns, and fostering nonjudgmental environment lead to collective responsibility for the cohort as well as the overall program. In essence, our approach espoused an informal reward/punishment mechanism without explicitly establishing it. Once we had the buy-in of all faculty members involved, the collective structure emerged.

Each cohort formed a task group of coworkers (Denson et al., 2006; Lickel, Schmader, and Hamilton, 2003). Within each cohort, collective responsibility manifested itself in two ways: 1) Responsibility by omission – each cohort member looked out for others and prevented them from deviating from the agreed upon program design agenda and 2) Responsibility by commission – each cohort member tended to perform to the best of their abilities to make sure that their cohort remained on the right track. The combination of these two sub-components (responsibility by omission and responsibility by commission) developed a self-correcting system with a tendency to achieve equilibrium, where equilibrium meant desired program development outcomes.

Although collective responsibility is often deemed as altruistic or prosocial behavior, an understated form of egoism still lingers in the behavior. As all faculty member had agreed to the shared program vision, any action that may be detrimental to the cohort and the program overall could jeopardize an individual faculty member's self-interest in the long run. Hence, cooperation, helping others, and self-correction were natural outcomes of this healthy competition between and among cohorts.

4.3.2 Peer monitoring: Although collective responsibility was the desired outcome of the cohort groups and the overall program, peer monitoring represents the underlying mechanism facilitating the development of collective responsibility. Peer monitoring refers to the observation of coworkers and reacting to it if the observed behavior was deemed inappropriate (Welbourne, Balkin, and Gomez-Mejia 1995; Welbourne and Farrante, 2008). In the context of program development, peer monitoring occurred when individual faculty member's actions appeared to drive the development process in an undesirable direction. Throughout the entire development process, there were multiple instances where peer monitoring allowed participants to exercise their collective responsibility. For example, there were multiple instances where an individual faculty member proposed changes to a course that seemed arbitrary. Cohort faculty in these instances stepped up and asked for a rationale to justify the change. In some cases, the change was accepted, and in others, it was not accepted. There were several occasions where faculty had to go back multiple times and reconsider their proposed changes based on feedback from cohort faculty. The impact of collective responsibility and peer monitoring is also evident from progression/change of course SLOs from Table 6 to Table 9. Note that some of the SLOs were modified as were some of the mappings. The iterative consensus-building process guided by collective responsibility and peer monitoring led to such a result.

A high degree of coordination and self-correction was possible as we made sure all faculty members teaching in the program understood not just the courses they teach, but also that they internalized program-level objectives and saw the connections between and among the courses in the program. When faculty members are involved in the program design process, the focus shifts from a concern of performance evaluation by external entities to one of mutual accountability about the group and its members. Assurance of learning then becomes more a matter of improving student learning than merely complying with the needs or requirements of external entities.

In our program development experience, involving groups of faculty led to higher levels of mutual accountability. For example, each faculty member had to explain the nature of activities and assessment in their class. Other members of a particular cohort essentially "peer-reviewed" these activities and assessments. This rigorous process ensured mutual accountability. Rashid (2015) suggests that teams "make timely performance adjustment[s]" when they hold each other mutually accountable. Each team member developed a keen understanding of the work being done. For example, if a faculty member changes a part of his or her course, it will be easy for others to understand the impact on their course and adapt accordingly in a timely fashion.

Lesson: Foster teamwork, collective responsibility, and peer-monitoring, which will lead to mutual accountability.

4.4 Creating a Learning Global Positioning System (GPS)

As we experienced, designing a high-quality and integrated graduate program requires alignment, faculty involvement, teamwork, and mutual accountability. Ensuring that the program design process focuses on student learning (i.e., help students see what they are learning and why) is equally important. In the study conducted by Light (2001) (as cited in Schmidt-Wilk, 2011), "Enhancing students' awareness of the big picture" goes beyond the specifics of a topic or course designed to help improve student learning. As we designed the MSIS program, we realized that we needed to create a program map that would help the students navigate and manage their learning. Students should be able to see the "big picture" as well as understand how individual courses are a part of the learning journey.

Students enrolling in a program often get their information about the program from either a website or a brochure that identifies relevant information about the program. This information is normally generated during the program design process and may include the following for a prospective student: courses, concentrations, career opportunities, contact information, and the like. Once admitted, the advisor guides the student on the path they need to traverse for fulfilling program requirements. When faculty collaboratively design an academic program, it is important to understand the path that the student will traverse to get to the endpoint in this program.

Beginning with the end in mind helped create what we call a "Learning GPS" for the proposed program. The Learning GPS provided students with a holistic view of the program and assisted them in understanding the path they will take to complete the program. The Backward Design approach facilitates the creation of a coherent Learning GPS as faculty members begin with the end in mind – the PLOs. This Learning GPS helps the students see their individual learning route, both the courses they will take in the program and the different paths (concentrations) they can follow to get to their learning goal (PLOs). Often, students are challenged in understanding why they are required to take a specific course and why it matters. The Learning GPS is a combination of Tables 1, 2, 3, 5, and 6. These tables can help visualize the courses and the interconnections between the courses in the program, helping the students understand why different courses are needed as well as how they tie into program-level objectives.

Faculty in a program are experts in specific areas and understand concepts and skills they teach and how they are related to concepts and skills in other courses in a program. Students normally learn concepts and skills in a course and may not necessarily connect them to other courses they take. There is a risk that this knowledge may exist in isolation in their minds and hence they question the value of different courses they are required to take. A Learning GPS simplifies the interconnections between different courses, and students can easily understand the value that each course adds to the program. It can help students in elucidating connections between the concepts they learn in different courses in a program. Further, the students can also make connections between what they learn in different courses and real-world events. Learning GPS can help students to 1) understand difficult and interconnected concepts improving their problem-solving skills and 2) map their learning to different career choices they want to pursue.

We argue that the Backward Design approach we used to design our MSIS program enabled us to develop a Learning GPS for students simultaneously. Faculty, working collaboratively, developed courses that depicted clear value for both faculty and students.

Lesson: Develop a Learning GPS to enhance student learning in a program.

5. CONCLUSION

Designing an academic program is a challenging task that is exacerbated when the program is being designed for 1) a discipline that is interdisciplinary and 2) delivery in a face-to-face and online mode. Our journey through the design process using the Backward Design model has taught us valuable lessons that can be used by other academic institutions. The context of program development (the program type, student body, and delivery mode) may differ across institutions; the insights can still be useful to other institutions seeking to employ Backward Design for program development.

To the best of our knowledge, this research is the first documented attempt to employ the Backward Design model in IS graduate program development. Further, our development approach built upon prior research in graduate IS program development, which explicitly called for the development of more "integrated IS programs" (Topi et al., 2016). In terms of future research, additional competency levels can be explored. In our work, we looked at whether a particular class developed/utilized competency in a specific domain (Individual foundational, IS core, and Domain) at the primary or secondary level. Tables 1, 2, and 3 explained these ideas. There are opportunities to explore competency levels in greater detail. Instead of suggesting that a particular class develops primary or secondary competency in a specific domain, each class can be broken into a series of competencies tied to roles that IS programs cater to (IT analyst, business analyst, etc.).

Our work on Backward Design can be further expanded to actively include industry partners. Recent research has indicated that there is potential for industry to get involved in program design (Mills, Chudoba, and Olsen, 2016; von Konsky, Miller, and Jones, 2016). It will be a worthwhile endeavor to take the structure of the Backward Design model and actively involve industry partners. Merging of novel and relevant frameworks like Skills Framework for the Information Age (SFIA) with a Backward Design approach can lead to a tractable, reproducible, industry-relevant, and transparent curriculum development process.

Additionally, this research can be expanded to include "Blended/Hybrid" options. We focused on face-to-face and online channels, and our singular aim was to keep a single set of course objectives, activities, learning experiences, and assessment criteria. Hybrid courses may face related but distinct challenges. For example, consider a student in a specific class, if he/she was not able to attend face-to-face sessions but contributed online or wanted to do so. Such a situation will pose not only an assessment conundrum for the instructor but also technological issues as it would be difficult for an online student to coordinate with his/her fellow students who were attending the session. Our program does not have a hybrid component; a student can either enroll in a fully online format or in a face-to-face format. Curricular issues in a blended program are an important research topic, but are outside the scope of this article.

Future research can also quantify the impact of program design. Surveys can be designed to garner evidence from various stakeholders (students, faculty, administration, and the industry). Triangulation of results can be used as the measure of effectiveness and success of the Backward Design approach.

Our study has documented a curricular development process used to create a graduate IS program. Although the principles of Backward Design are not new, their application to a graduate IS program is relatively unique. The program developed using this model helped align it with the vision and mission of our college and facilitated faculty engagement. The process has resulted in a truly integrated program where courses interrelate and build upon each other. Furthermore, engaged faculty became more accountable to the program resulting in a unique learning opportunity for students.

Using Backward Design also allowed us to realize benefits of AOL standards recommended by AACSB. These standards emphasize *engagement, innovation,* and *impact.* Involving all faculty members in the program development effort enabled *engagement*. Our application of Backward Design principles is an *innovative* approach in IS graduate program design. The new graduate program helped faculty to see the forest for the trees, which is *impactful* given the individual proclivities of faculty members. We anticipate the impact on student learning to be favorable as assessments were built into the program as a feature of program design.

6. REFERENCES

Aasheim, C., Shropshire, J., Li, L., & Kadlec, C. (2012). Knowledge and Skill Requirements for Entry-level IT Workers: A Longitudinal Study. *Journal of Information Systems Education*, 23(2), 193-204.

AACSB. (2007). Eligibility Procedures and Accreditation Standards for Business Accreditation. Tampa, FL: AACSB International.

Abraham, S. (2006). Assessing the Learning Outcomes of a Computer Information Systems (CIS) Program. *Information Systems Education Journal*, 4(14), 1-6.

Abdous, M. H. & Yoshimura, M. (2010). Learner Outcomes and Satisfaction: A Comparison of Live Video-Streamed Instruction, Satellite Broadcast Instruction, and Face-to-Face Instruction. *Computers & Education,* 55(2), 33-741.

Appana, S. (2008). A Review of Benefits and Limitations of Online Learning in the Context of the Student, the Instructor, and the Tenured Faculty. *International Journal on ELearning*, 7(1), 5-22.

Allen, I. E. & Seaman, J. (2015). *Grade Level: Tracking Online Education in the United States*. Babson Park, MA: Babson Survey Research Group. Retrieved March 10, 2015, from http://www.onlinelearningsurvey.com/reports/gradelevel.pdf.

Allen, I. E. & Seaman, J. (2010). *Class Differences: Online Education in the United States, 2010*. Sloan Consortium (NJ1). Retrieved from http://files.eric.ed.gov/fulltext/ED529952.pdf.

Bell, C. C., Mills, R. J., & Fadel, K. J. (2013). An Analysis of Undergraduate Information Systems (IS) Curriculum: Adherence to IS 2010 Curriculum Guidelines. *Communications of the Association for Information Systems*, 32(2).

Bybee, R. (2006). Enhancing Science Teaching and Student Learning: A BSCS Perspective. Boosting Science Learning: What it will Take. *ACER Research Conference. Review of Educational Research*, 64, 1–35.

Chiang, R. H., Goes, P., & Stohr, E. A. (2012). Business Intelligence and Analytics Education, and Program Development: A Unique Opportunity for the Information Systems Discipline. *ACM Transactions on Management Information Systems (TMIS),* 3(3), Article 12.

Childre, A., Sands, J. R., & Pope, S. T. (2009). Backward Design: Targeting Depth of Understanding for All Learners. *Teaching Exceptional Children,* 41(5), 6-14.

Chong, D. Z., He, W., & Wu, H. (2012). Leveraging Cloud Computing to Support Experiential Learning in Distance Education. *IEEE Technology and Engineering Education*, 7(2), 17-19.

Costigan, R. D. & Brink, K. E. (2015). Another Perspective on MBA Program Alignment: An Investigation of Learning Goals. *Academy of Management Learning & Education*, *14*(2), 260-276.

Couger, J. D., Davis, G. B., Dologite, D. G., Feinstein, D. L., Gorgone, J. T., Jenkins, A. M., Kasper J. C. L., Longenecker, H. E., & Valacich, J. S. (1995). IS'95: Guideline for Undergraduate IS Curriculum. *MIS Quarterly*, 19(3), 341-359.

Daugherty, K. K. (2006). Backward Course Design: Making the End the Beginning. *American Journal of Pharmaceutical Education*, *70*(6), 135.

Denson, T. F., Lickel, B., Curtis, M., Stenstrom, D. M., & Ames, D. R. (2006). The Roles of Entitativity and Essentiality in Judgments of Collective Responsibility. *Group Processes & Intergroup Relations*, 9(1), 43-61.

Diefenbach, T. & Sillince, J. A. (2011). Formal and Informal Hierarchy in Different Types of Organization. *Organization Studies*, 32(11), 1515-1537.

Ducrot, J., Miller, S., & Goodman, P. S. (2008). Learning Outcomes for a Business Information Systems Undergraduate Program. *Communications of the Association for Information Systems*, 23(1), 96-122.

Gorgone, J. T., Gray, P., Stohr, E. A., Valacich, J. S., & Wigand, R. T. (2006). MSIS 2006: Model Curriculum and Guidelines for Graduate Degree Programs in Information Systems. *ACM SIGCSE Bulletin*, 38(2), 121-196.

Gupta, B., Goul, M., & Dinter, B. (2015). Business Intelligence and Big Data in Higher Education: Status of a Multi-Year Model Curriculum Development Effort for Business School Undergraduates, MS Graduates, and MBAs. *Communications of the Association for Information Systems*, 36(23), 450-474.

Guidry, B. N., Stevens, D. P., & Totaro, M. W. (2011). The Systems Analysis and Design Course: An Educators' Assessment of the Importance and Coverage of Topics. *Journal of Information Systems Education*, 22(4), 331-345.

Hamilton, V. L. (1978). Who is Responsible? Toward a Social Psychology of Responsibility Attribution. *Social Psychology,* 41(4), 316-328.

Havelka, D. & Merhout, J. W. (2009). Toward a Theory of Information Technology Professional Competence. *Journal of Computer Information Systems*, 50(2), 106-116.

He, W. & Yen, C. J. (2014). The Role of Delivery Methods on the Perceived Learning Performance and Satisfaction of IT Students in Software Programming Courses: A Case Study. *Journal of Information Systems Education*, 25(1), 23-34.

Hepner, M. & Dickson, W. (2013). The Value of ERP Curriculum Integration: Perspectives from the Research. *Journal of Information Systems Education*, 24(4), 309-326.

Jacobi, F., Jahn, S., Krawatzeck, R., Dinter, B., & Lorenz, A. (2014). Towards a Design Model for Interdisciplinary Information Systems Curriculum Development, as Exemplified by Big Data Analytics Education. *Proceedings of the European Conference on Information Systems (ECIS)*, Tel Aviv, Israel.

Kim, D. J., Yue, K. B., Al-Mubaid, H., Hall, S. P., & Abeysekera, K. (2012). Assessing Information Systems and Computer Information Systems Programs from a Balanced Scorecard Perspective. *Journal of Information Systems Education*, 23(2), 177-192.

Krathwohl, D. R. (2002). A Revision of Bloom's Taxonomy: An Overview. *Theory into Practice*, 41(4), 212-218.

Kumar, A., Shah, V., & Smart, K.L. (2017). Moving Forward by Looking Backward: Using Backward Design to Develop an Innovative MSIS Program. *Proceedings of the 23rd Americas Conference on Information Systems.*

Lee, D. M., Trauth, E. M., & Farwell, D. (1995). Critical Skills and Knowledge Requirements of IS Professionals: A Joint Academic/Industry Investigation. *MIS Quarterly*, 19(3), 313-340.

Lickel, B., Schmader, T., & Hamilton, D. L. (2003). A Case of Collective Responsibility: Who else was to Blame for the Columbine High School Shootings? *Personality and Social Psychology Bulletin*, 29(2), 194-204.

Light, R. (2001). *Making the Most of College: Students Speak their Minds*. Cambridge, MA: Harvard University Press.

Luke, A., Woods, A., & Weir, K. (Eds.). (2013). *Curriculum, Syllabus Design, and Equity: A Primer and Model*. Routledge.

May, J. & Lending, D. (2015). A Conceptual Model for Communicating an Integrated Information Systems Curriculum. *Journal of Computer Information Systems*, 55(4), 20-27.

McTighe, J. & Thomas, R. S. (2003). Backward Design for Forward Action. *Educational Leadership*, 60(5), 52-55.

Mills, R. J., Chudoba, K. M., & Olsen, D. H. (2016). IS Programs Responding to Industry Demands for Data Scientists: A Comparison between 2011-2016. *Journal of Information Systems Education*, 27(2), 131-140.

O'Neill, G. (2010). Programme Design: Overview of Curriculum Models. Retrieved from https://www.ucd.ie/t4cms/UCDTLP00631.pdf.

Patten, K. P. & Harris, M. A. (2013). The Need to Address Mobile Device Security in the Higher Education IT Curriculum. *Journal of Information Systems Education*, 24(1), 41-52.

Pratt, J. A., Keys, A., & Wirkus, T. (2014). Preparing Information Systems Graduates for a Complex Society: Aligning IS Curricula with Liberal Education Learning Outcomes. *Journal of Information Systems Education*, 25(1), 35-44.

Rashid, F. (2015). Mutual Accountability and its Influence on Team Performance. Cambridge, MA: Harvard University Press.

Sanders, J., Hamilton, V., Denisovsky, G., Kato, N., Kawai, M., Kozyreva, P., Kubo, T., Matsovsky, M., Nishimura, H., & Tokoro, K. (1996). Distributing Responsibility for Wrongdoing inside Corporate Hierarchies: Public Judgments in Three Societies. *Law & Social Inquiry*, 21(4), 815-855.

Schmidt-Wilk, J. (2011). Course Design as a Strategic Process. *Journal of Management Education*, 35(3), 319-323.

Swan, K., Day, S. L., Bogle, L. R., & Matthews, D. B. (2014). A Collaborative, Design-Based Approach to Improving an Online Program. *The Internet and Higher Education*, 21, 74-81.

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, J. F., Jr., Sipior, J. C., & de Vreede, G. J. (2010). IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. *ACM/AIS Task Force*

Topi, H., Donnellan, B., Ramesh, V., Conboy, K., Van Toorn, C., & Wright, R. (2014). Moving Toward the Next Generation of Graduate Degree Programs in Information Systems. *Communications of the Association for Information Systems*, 34(35), 693-710.

Topi, H., Kartsen, H., Brown, S., Carlvalho, J. A., Donnellan, B., Shen, J., Tan, B., & Thouin, M. (2016). MSIS 2016: Global Competency Model for Graduate Degree Programs in Information Systems. *ACM/AIS Task Force*.

US News. (2016). Retrieved from https://www.usnews.com/education/online-education/articles/2016-02-09/study-enrollment-in-online-learning-up-except-at-for-profits.

Veltri, N. F., Webb, H. W., Matveev, A. G., & Zapatero, E. G. (2011). Curriculum Mapping as a Tool for Continuous Improvement of IS Curriculum. *Journal of Information Systems Education*, 22(1), 31-42.

von Konsky, B., Miller, C., & Jones, A. (2016). The Skills Framework for the Information Age: Engaging Stakeholders in ICT Curriculum Design. *Journal of Information Systems Education*, 27(1), 37-50.

Watty, K., McKay, J., & Ngo, L. (2016). Innovators or Inhibitors? Accounting Faculty Resistance to New Educational Technologies in Higher Education. *Journal of Accounting Education*, 36, 1-15.

Welbourne, T. M. & Ferrante, C. J. (2008). To Monitor or not to Monitor: A Study of Individual Outcomes from Monitoring One's Peers under Gainsharing and Merit Pay. *Group & Organization Management*, 33(2), 139-162.

Welbourne, T. M., Balkin, D. B., & Gomez-Mejia, L. R. (1995). Gainsharing and Mutual Monitoring: A Combined Agency-Organizational Justice Interpretation. *Academy of Management Journal*, 38(3), 881-899.

Wiesenberg, F. & Stacey, E. (2005). Reflections on Teaching and Learning Online: Quality Program Design, Delivery and Support Issues from a Cross-Global Perspective. *Distance Education*, 26(3), 385-404.

Wiggins, G. P. & McTighe, J. (2001). What is Backward Design? In *Understanding by Design* (pp. 7-19). Upper Saddle River, NJ: Merrill Prentice Hall.

Wiggins, G. P. & McTighe, J. (2005). *Understanding by Design*. Association for Supervision and Curriculum Development (ASCD).

Winch, C. (2013). Curriculum Design and Epistemic Ascent. *Journal of Philosophy of Education*, 47(1), 128-146.

Woodward, B., Imboden, T., & Martin, N. L. (2013). An Undergraduate Information Security Program: More than a Curriculum. *Journal of Information Systems Education*, 24(1), 63-70.

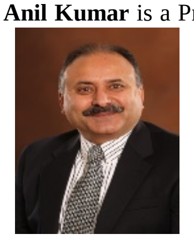
Yamagata-Lynch, L. C., Cowan, J., & Luetkehans, L. M. (2015). Transforming Disruptive Technology into Sustainable Technology: Understanding the Front-End Design of an Online Program at a Brick-and-Mortar University. *The Internet and Higher Education*, 26, 10-18.

Yin, R. (1994). *Case Study Research: Design and Methods*. Beverly Hills, CA: Sage Publications.

Yang, Y. (2010). Roles of Administrators in Ensuring the Quality of Online Programs. *Knowledge Management & E-Learning: An International Journal (KM&EL)*, 2(4), 363-369.

AUTHOR BIOGRAPHIES

Vishal Shah is an Assistant Professor of Information Systems at Central Michigan University. His research interests include IS pedagogy, social media, IT and learning, game-based learning, and telemedicine. He has published in journals such as *AIS-Transactions on Replication Research*, *International Journal of Telemedicine Applications*, *Journal of Education for Business*, and *Technical Communication Quarterly* as well as in leading IS conferences such as AMCIS, IACIS, and DSI.



Anil Kumar is a Professor of Information Systems at Central Michigan University. He is a GITMA fellow and the 2018–2019 Towle Professor of Information Systems at Central Michigan University. His research interests include digital transformation of organizations, managing information resources, technology-mediated learning, and creating value from social media technologies. He has published in *Decision Sciences Journal of Innovative Education, American Journal of Business*, *Journal of Global Information Technology Management*, *Decision Support Systems*, *The TQM Journal*, *Total Quality Management and Business Excellence*, *Industrial Management and Data Systems*, and *Information Processing and Management*. Dr. Kumar has presented more than 50 papers at international and national conferences.

Karl L. Smart is Associate Dean in the College of Business Administration at Central Michigan University and former Chair of the Department of Information Systems. His areas of research interest include information and document design, the impact of technology on communication and the workplace, collaborative learning and working in teams, and assurance of learning. He has published widely in professional and business communication journals, including *International Journal of Business Communication*, *Business and Professional Communication Quarterly*, *IEEE Transactions on Professional Communication*, *Technical Communication*, *Journal of Business and Technical Communication*, as well as articles in information systems and education/pedagogy journals.

Information Systems & Computing
Academic Professionals

EDSIG
Serving Information Systems Educators

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 2574-3872