Moving Forward by Looking Backward: Using Backward Design to Develop an Innovative MSIS Program

Full Paper

Anil Kumar
Central Michigan University
kumar1a@cmich.edu

Vishal Shah
Central Michigan University
shah3v@cmich.edu

Karl Smart
Central Michigan University
smart1kl@cmich.edu

Abstract

Program design is a challenging task that involves significant effort and resources. The challenge is compounded if the program is being designed for an interdisciplinary discipline. Multiple approaches are available for program designers, and varying approaches may work well in different contexts and settings. The fundamental core of Backward design is that the selected approach may be successfully applied in different contexts if it begins with the end in mind. This research describes how a Midwestern public university used backward design to construct a curriculum process that gives voice to the expertise, proclivities, and preferences of individual faculty members in curriculum development while aligning course objectives with the program’s mission and vision. As a major contribution, the paper shares lessons learned from a collaborative program design effort using the principles of Backward design. This effort was undertaken to design the Masters of Science in Information Systems (MSIS) program at a Midwestern public research university.

Keywords: Backward design, IS curriculum, graduate.

Introduction

As information systems (IS) educators, our primary concern is to design a curriculum that is of value to our students. Value for business school students tends to translate as job-readiness for graduates. Preparing graduates for the IS field is a significant challenge, given its interdisciplinary nature and ever-changing skill set. The interdisciplinary nature of IS programs is highlighted by prior research (Ducrot, Miller, and Goodman 2008; Havelka and Merhout 2009; Aasheim, Shropshire, Li, and Kaldec 2012). Board consensus among IS researchers shows that IS graduates need to have an adequate set of technical skills, communication and teamwork skills, and domain knowledge specific to the business area (Aasheim et al. 2012; Havelka and Merhout 2009).

Considerable research focuses on the method by which an IS program imparts necessary skills to graduates. Multiple examples of studies concentrate on learning outcomes for an information system (IS) program (Ducrot, Miller, and Goodman 2008; Abraham 2006), although most of these studies focus on undergraduate programs. IS curriculum development has been a vibrant research area, but most of the curriculum research is done at the undergraduate level and is based on model IS curriculum (Bell, Mills, and Fadel 2013). Scant research exists focused on graduate program/curriculum development. Such paucity provides challenges for graduate programs being developed, especially those delivered in an online format. This area has begun to become a focus of curriculum development research.

Program design is a challenging process to begin with, and when a program is being designed both for online and for face-to-face delivery, the challenges multiply. Research indicates that the quality of online business education (Abdous and Yoshimura 2010; Yang 2010) is a serious concern. Recognizing these challenges, the Association to Advance Collegiate Schools of Business (AACSB—the leading accreditation
organization for business schools) formed a task force in 1999 to address issues in distance learning; an additional task force updated the report in 2007, providing specific suggestions. The 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).

Extant research in curriculum development provides “models [which] help designers to systematically and transparently map out the rationale for the use of particular teaching, learning and assessment approaches” (O’Neill 2010). This outcome-based approach has been very successful in designing curriculum at an undergraduate level. One should note, however, that there is considerable variety in successful IS undergraduate curriculum, often with a significant departure from IS model curriculum (Topi, Valacich, Wright, Kaiser, Nunamaker Jr, Sipior, and de Vreede 2010). This variety affirms the interdisciplinary nature of the field. This range of curriculum necessitates that the process of program development is streamlined, with integration driving the process of development. Actual curriculum models may vary based on the context in an institution.

Although much of today’s curriculum was developed for traditional face-to-face delivery, recent changes in economy and technology have been significant factors in pushing education to online delivery (Allen and Seaman 2015). Educational institutions and educators need to adapt to the changing requirements of the students and society. Overall, there has been an increase in the number of working professional who wish to enroll and complete graduate-level education, and enrollment in online courses and programs has climbed for a number of years (Allen and Seaman 2015; OLC 2014; US News 2016). This trend is likely to hold for graduate degrees in the IS field as well, with significant earning and growth potential in the discipline (Payscale, 2015)

One of the key benefits of online programs comes from students’ ability to now earn degrees without having to go to a specific physical university campus location that may be a considerable distance from the workplace; this allows them to continue their employment while furthering their education (Chong et al. 2012; He and Yen 2014). The shift in learning needs and the changing demographics of graduate students, combined with the limited research on graduate level IS program development, present an ideal opportunity for IS research to advance a method of IS graduate education, especially in the area of online programs. As IS educators, we need to rethink the ways in which we design and deliver courses. This paper details one such attempt at program design at the graduate-level, applicable to both online and face-to-face audiences to ensure accomplishment of learning outcomes.

Specifically, we apply the framework of “Backwards design” to IS curriculum development. This research is focused on answering the following two questions:

1. How do we, as IS educators, construct a curriculum process that gives voice to expertise, proclivities, and preferences of the individual faculty members, while still aligning with course objectives and program, college, and university missions and visions?

2. Can we deduce general curriculum design principles from the curriculum development process mentioned above?

Building upon a foundation of curriculum development and changing education variables, the next section describes the state of research in curriculum development in the field of Information Systems (IS). We then describe the Backward design process prevalent in curriculum design and its application. The article ends with a discussion of lessons learned and a set of recommendations for replicating the curriculum development procedure.

**Literature Review**

Information Systems (IS) is a multidisciplinary field (Aasheim et al. 2012; Havelka and Merhout 2009). The rapidly changing technological landscape is constantly re-shaping the IS field. The Venn diagram (Figure 1) represents the three types of skills that constitute an ideal IS graduate is expected to possess. Employers used to seek IS graduates primarily due to technical skills, but the repertoire of competencies needed to gain employment upon graduation has expanded beyond just technical skills; employers are seeking graduates who are adept at problem-solving and have a range of business and technical skills. IS curriculum can be designed to provide technical and communication skills but IS graduates tend to gain
business skills on the job. Although, certain elective coursework in diverse fields, such as finance, HR, and accounting, can help but cannot act as a substitute for domain skills. Technical and communication skills, on the other hand, are relatively specific and can be packaged in a variety of courses.

Surendra and Denton (2009) state “a basic task for educators and administrators in IS programs is to design a curriculum that provides value for their students” (p. 78). The value in the case of IS students primarily in business schools translates to a combination of technical and communication skills enabling them to secure employment. In response to the changes in the labor market, IS programs have begun to deliver interdisciplinary education at the intersection of computer science, business, and communication studies (Jacobi, Jahn, Krawatzeck, Dinter, and Lorenz 2014).

A variety of combinations of these three fields can produce a set of viable IS curricula. Despite the existence of the model IS Curriculum (Topi et al. 2010), compliance with Model IS curricula is very low (Choi, Ulema, and Waldman 2008; Apigian and Gambill 2010). The uneven compliance appears to be a symptom of the variety of IS curriculums offered. About 48 percent of the surveyed programs complied with model IS curriculum. This compliance rate reflects the variety in IS and needs of the employers. Various career tracks beyond those mentioned in model IS curriculum also enjoyed a favorable reception across education institutes (Bell et al. 2013).

As we discussed, various types IS curricula exist across AACSB business schools. In retrospect, these curricula can qualitatively be categorized into the following categories.

**Learner-centered:** This strand of research is based on constructivism (Driver and Oldham 1986; Thompson 2001). According to constructivism, the learner constructs his/her knowledge based on the context and interaction with the instructional material. Prior IS curriculum has advocated building a curriculum in this vein (Chen 2003; Chen, Shaw, and Yang 2006; Rubens, Kaplan, and Okamoto 2010).

**Subject-centered:** In this category, the emphasis on the specific topics drives the collective consensus of the faculty. Faculty then design a set of interconnected classes and disseminate information via a conceptual model (May and Lending 2015). While this type of model takes into account stakeholders such as industry and the organization, the most important factors such as learning objectives for students does not seem to play a major role in the model development process.

**Problem-centered:** Research in this sub-area focuses on specific problems and merits of problem-based learning (Savery and Duffy 1995). The main idea is that engagement with an experiential problem will enhance student learning (Bentley 1999; Woodward, Sendall, and Ceccucci 2010).

A variety of curriculum models can be described along a continuum of a broader category called product and process models (O’Neill 2010). As the name suggests, “product model” is focused on the structure and emphasizes plans and activities. “Process models,” on the other hand, have more flexibility and emphasize learning rather than adhering to specific plans. Note that every curriculum has implemented various parts of each category discussed above, and as such is difficult to isolate as a stand-alone island with clear boundaries. Figure 2 shows the landscape of IS curriculum research.

---

**Figure 1. Skillset of an Ideal IS Graduate**

---

**Figure 2. The Landscape of IS Curriculum Research**

---

*Twenty-third Americas Conference on Information Systems, Boston, 2017*
Generally, a specific IS curriculum will have various degrees of the three approaches mentioned above at varying levels of product and process orientation. IS curriculum research has been done at both undergraduate and graduate levels (Couger, Daniel, Davis, Dologite, Feinstein, Gorgone, Jenkins, Kasper, Little, Longenecker Jr, and Valacich 1996; Topi et al. 2010). Research on graduate-level IS program development is scant (Jacobi et al. 2014; Gorgone, Gray, Stohr, Valacich, and Wigand 2006). Having described the state of curriculum development in IS, the following limitations can be addressed:

1. Most curriculum design research is based on the subject-centered approach (Noll and Wilkins 2002; May and Lending 2015). However, it does not include the documented communication between course designers that took place to devise the model, nor have any prescribed general principles been articulated to achieve the same curriculum outcomes. In other words, the conceptual models were provided based on subject-centered approach and on alumni surveys, but in many cases, the models were not applied to actual course development, nor was the process of development clearly documented. In a recent paper, May and Lending (2015) developed a conceptual model for IS curriculum development and applied it to formulate a set of courses, but the “process flow” required for such a development was not apparent from the article.

2. Frieder et al. (2014) provide a reproducible template based on Business Process Modeling Notation (BPMN), which generates a course catalog based on Bloom’s taxonomy. The template is highly useful and reproducible, but it approaches the problem from the ontological point of view and provides a solution rooted in BPMN and Unified Modeling Language (UML). While this is a very useful and productive solution to a pressing research problem, it does not take into account various environmental constraints, such as specific faculty or department idiosyncrasies. As Frieder asserts: “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...” (p. 13). Thus, there is a possibility for developing and documenting IS curriculum taking into account institute-related constraints in a non-parameterized way, as well as keeping the developmental process modular for other organizations/institutes to adopt it readily.

3. Recent research has tackled the issue of graduate curriculum design (Gupta, Goul, and Dinter 2013) in specific areas such as Business Intelligence based on Krathwohl’s (2002) revised taxonomy. While immensely useful, the model curriculum is suitable to one sub-field of IS (i.e., Business Intelligence). This opens an opportunity to develop a generic IS program that may then have more than one concentration.

4. Most IS curricula research has strived to develop courses to meet a set of consensus skills (Jacobi et al. 2014). These skills are often based on the key points derived from prior curriculum development research (Lee, Truth, and Farwell 1995; Chiang, Goes, and Stohr 2010), as well as triangulation of the stakeholders (i.e., alumni, employers, and academic institutions). A very recent example in this vein is the development of Skills Framework for the Information Age (SFIA) (Konsky, Miller, and Jones 2016). SFIA promises to be a very productive framework as it facilitates interaction between faculty and the industry. Instructors design their courses to meet...
the agreed upon end goals and course objectives; however, such an approach may not lead to the desired curriculum. Some reasons why the disparity can happen are listed below:

- Often faculty members work in isolation to develop their courses. While content in the course may be in alignment with the course objectives, the faculty may not clearly appreciate or understand how their course fits with other courses. This points to the importance of the communication piece in the curriculum-development puzzle. We found no significant research documenting the process of coordination and communication among various faculty members during curricular program development. In other words, the process flow that faculty used to develop the curriculum was not apparent.

- Given the expertise, proclivities, and priorities of individual instructors/faculty members, the curriculum development process does not lend itself to standardization. While there have been attempts to automate the entire process (Topi et al. 2014), these attempts approach curriculum from a design-science point of view. We approach the same problem from a behavioral and communications standpoint.

- Earlier, we mentioned that coordination among faculty members is crucial to develop an effective IS program. Equally important is to document these practices as well as generalize these practices. Some prior research (Kim, Yue, Al-Mubaid, Hall, and Abeysekera 2012) has adopted a process-based approach to develop and assess IS programs. However, we found that guidelines on reproducing these practices were specific to curriculum assessment and to ABET guidelines. As general design practices are not well-known or documented, it may impact the overall quality of an IS program and those who wish to customize the proposed IS curriculum for specific purposes. Most of the curriculum development work is done at an undergraduate level. Given current push towards executive and graduate education, we must investigate graduate level-IS curriculum development in more detail, with both online and face-to-face modes.

Our approach to curriculum development is at program-level, keeping in mind the institution’s mission and values; the approach follows a process-based philosophy. As we have seen, there are gaps in graduate curriculum development literature focused at a program-level and addressing both modes of delivery (face-to-face and online). Much of the prior research has also been scattered and ad-hoc in nature. We build on prior research to fill this gap while providing a set of reproducible recommendations based on a widely used curriculum development framework called “Backwards Design.” Specifically, we build on the recent framework provided by recent research on IS graduate programs (Topi et al. 2014; AIS 2016). The framework provides flexibility to contextualize the development of graduate IS programs based on institutes. The generic process starts from considering course objectives and arrives at course content and pedagogy (Topi et al. 2014). We instantiate this process via the use of Backward design. The next section describes the Backward design process we used in developing our MSIS curriculum.

**Backward Design**

Backward design (Wiggins and McTighe 2001; 2005) refers to a shift in curriculum design in which instructional designers focus on desired learning outcomes. As curriculum designers work backward from the desired end, they decide on the acceptable evidence for such learning, along with the kind of activities and content that would be appropriate to achieve the desired outcomes. The approach has three stages. The first stage is to identify of desired outcomes. The second stage is to determine acceptable evidence that indicates achievement of desired outcomes. The final stage is to select appropriate content/lesson plans and contents that are germane to the defined result and acceptable evidence.

Outcomes-based curriculum design is not new in IS (O’Neill2010). However, the research following in this tradition has been largely theoretical and scattered, not rooted in a well-established model. This research works to fill that void. It has to be noted that this work is exploratory in nature, though it answers the “why” and “how” questions (Yin 1994, p. 6). We believe that the process of curriculum development provides insights about why Backward design is an appropriate approach and how we implemented it. In addition, the lesson learned can be contextualized by other institutes. As we applied the Backward design model to a set of interrelated courses, we paid specific attention to faculty interests and communication. Communication between various faculty and its inclusion in curriculum design is critical to success, an
issue that has not been the focus prior IS research. By involving faculty members at each stage of the design and carefully mapping and documenting the interdependencies, we created a coherent combination of classes. To achieve coherency, we created various course groups. Each group belonged to a specific theme (such as Communications or Analytics or IS Core) and had a designated cohort coordinator. Each group met weekly to discuss desired outcomes, evidence, and learning activities. Cohort coordinators from various thematic groups also met regularly with the program director who coordinated the entire process. This approach helped in the mapping of interdependencies as well as accounting for individual faculty member preferences while facilitating mutual learning for the faculty involved in the process. Such an approach to design a robust “learning system” in a constantly changing environment has been recommended by Wals et al. (2009). Backward design espouses these principles of adaptive learning processes to create a sustainable learning system. Hence, Backward design was deemed as an appropriate approach for IS curriculum development. Using Backward design, we were able to create an integrated curriculum that encapsulated multiple viewpoints in a constantly changing landscape. Also, we clearly documented the process of design and culminated with a set of reproducible recommendations that can be adopted by other educational organizations. The next section describes the implementation process.

Implementing Backward Design

To manage the process of designing the program using principles of Backward design, we created an ad-hoc organizational structure. A program coordinator was assigned to the overall program, and we divided the program course offerings into three cohorts and assigned a coordinator for each cohort. The cohorts were designed based on a grouping of classes that had a potential to form a specific concentration, such as Enterprise Systems Cohort, Business Intelligence cohort, and Project Management cohort. The program coordinator met with cohort coordinators on a regular basis to ensure that work continued at a steady pace. Cohort coordinators worked with individual faculty in the cohort on a weekly basis.

Figure 3

**Backward design for designing MSIS program at Midwestern University**

The MSIS program design using Backward design principles was completed in three phases (Figure 3)—outcomes, assessment, and design. In the first phase (Table 1) of the program redesign using Backward design, all faculty members identified what they wanted students to understand, to know, and to be able to do in the course taught. These objectives were identified for each course, keeping the MSIS Program Learning Objectives (PLOs) in mind to provide clarity in terms of how each course aligned with the program. Keeping the program objectives in mind helped faculty see how the learning in their course related to and aligned with the concepts and skills students needed in the context of the program as a whole. The deliverable of this phase consisted of modified Student Learning Objectives (SLO’s) of each specific class. The cohort coordinator made sure that these SLOs were in overall agreement with the objectives at a cohort level, which were in turn derived from PLOs.

In the second phase, faculty determined acceptable evidence of learning for each course to ensure that student learning objectives were correctly assessed. The main goal was to clearly identify appropriate assessments for each course and make certain they aligned with the stated objectives of the course. With assessments tied to course objectives and course objectives aligned with PLOs, the mapping of PLOs for program-level assessment becomes apparent. The deliverable for each class in a specific cohort was a listing of the type of assessment appropriate for the concerned class in a specific cohort. For example, the faculty might decide that a student can be assessed via quizzes, a qualitative test, a project or presentation, or any combination of such measures. In doing so, the concerned faculty had to justify the type of assessment chosen.

In the third phase of the redesign process, faculty developed instructional strategies and learning activities for their courses. The instructional strategies and learning experiences help achieve learning
goals as captured by the assessment evidence. Taken together, the three phases translate the process described by AIS (2016) into practice. A significant advantage to this methodical approach is that learning activities within each cohort, and by extension among different cohorts, were not ad-hoc but tied to the SLOs of each course which in turn were pegged onto PLOs. Thus, various learning activities among and across cohorts complemented each other. A learning activity could vary across a broad range of spectrum such as business simulation, online forums to a discussion board. ¹

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outcomes: Identify desired results of MSIS program</td>
<td>Assessment: Determine acceptable evidence for MSIS program</td>
<td>Design: Plan learning experiences and instruction in MSIS courses</td>
</tr>
<tr>
<td>What is it that we want students to understand, know and be able to do?</td>
<td>How will we know that they will know what we want them to know?</td>
<td>What do we need to do in the classroom to prepare them for the assessment?</td>
</tr>
</tbody>
</table>

Table 1: Phases of Backward Design

Faculty members coordinated with each other and completed the MSIS program redesign process using Backward design in 12 weeks. At the end of the process, the program coordinator wrote a report that was distributed to all the faculty and department chair involved in the design process. The report served to seek feedback and identify any omissions in the process. Feedback from faculty was incorporated, and the program curriculum was sent through the university curriculum process. Using the design process described, we learned many things that may apply to other institutions seeking to revise their curriculum or better assess whether intended student learning goals are met. Among the key takeaways from the process are several crucial behaviors and activities that are discussed in the next section.

Lessons Learned and Discussion

Applying the Backward design approach to curriculum development, we learned several different lessons that may be useful for other programs.

Alignment

Alignment happens at multiple levels when designing an academic program. Care must be taken to ensure that PLOs are aligned with college vision and mission on the one hand and with student learning objectives at the course level (SLOs) on the other hand. This ensures that alignment exists between program vision and individual faculty vision, as well as assessment and learning experiences in individual courses. This facilitates the development of a broader educational vision that can be communicated more easily to stakeholders and can influence curriculum development. 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.”

Faculty involvement

Involving faculty members who teach in a program in the design process ensures individual faculty members understand the role their course plays in a program. Often faculty may create and teach a course that is well-designed, though it may not fit in with the overall program. Involvement in the design process ensures that individual faculty members understand the big picture as well as the specific focus of a course they teach. Involved and participating faculty working together to design a program are motivated as they perceive greater control of the design process of not just their course, but the entire program. The

¹ Please note we have detailed documentation of faculty rationale for each phase, as well as the mapping matrices that were derived from weekly meetings between program coordinator and cohort leaders. However, it is not feasible to present it in this article due to space constraints.
faculty in this process can be compared to the learner, where involvement in decision-making is perceived as a learner-centered approach to learning.

**Teamwork leading to mutual accountability**

An academic program includes multiple courses that are highly interdependent, and the success of the program depends on how well the faculty teaching in the program understand not just the courses they teach, but also the interconnections between the courses. 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 pertaining to the group and its members. Assurance of learning then becomes more a matter of improving student learning than complying with the needs or requirements of external entities. All faculty members on the team understand not only what they are responsible for, but also how their course fits in with other courses in the program. Rashid (2015) suggests that teams “make timely performance adjustment[s]” when they hold each other mutually accountable. Each team member develops 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.

**Focus on student learning**

Faculty care about student learning, and focusing on student learning during the program design process promotes an engaging and collaborative environment for faculty. As faculty sharing learning experiences, they develop their course to enhance student learning; this encourages reflection and exchange of ideas to enhance learning in the program.

Similarly, while individual faculty may develop good learning objectives for their courses, they may be frustrated when SLOs for their courses are viewed from the perspective of program assessment externally forced upon a program. In such a situation, faculty may seek guidance on the process of developing assurance of learning that enhances student learning, and assessment becomes a part of the development process, not an external requirement later forced upon the program.

**Encourage and support faculty to innovate**

Individual faculty members know the course they teach. They understand how it has evolved over the years, and they know what students expect from the course. Over time, they may try different things, introducing new learning experiences while trying to innovate in their courses. Often, different faculty members in the program may have a different view or mindset on these issues pertaining to a course or to multiple courses. Embracing different viewpoints of faculty and connecting the dots helps create an environment that promotes innovation. A department chair or the person leading the program design initiative should encourage and support creating such an environment. For example, transparency in communication and collective decision making making ameliorate trust in the group, which promotes the creation of an innovative environment. When faculty members hear different viewpoints and the reasoning behind the viewpoints, old models and mindsets are challenged, and new ones emerge. Shared mental models develop. Taking leaps of faith is a lot easier when there is trust, and this leads to innovation as new ideas emerge.

**Identify gaps in curriculum to create a seamless and integrated experience for students**

Designing a program with committed colleagues makes it easier to identify gaps in the curriculum. With shared commitment, it is a lot easier to see where a program may be missing key concepts or skills that provide an integrated and seamless learning experience for students. For example, two different instructors responsible for teaching a specific skill, such as data modeling, may approach it differently in their classroom. Understanding why and how a faculty member approaches a concept from a different perspective helps identify gaps. It is easy to address gaps in the curriculum when they are known.
Create a Global Positioning System (GPS) for learning

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. 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 helps create what we call a “learning GPS” for the proposed program. The learning GPS should provide a holistic view of the program and assist the student in understanding the path they will take to complete the program. The Backward design approach (Figure 3) facilitates the creation of a coherent learning GPS as faculty members begin with the end in mind, i.e., 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).

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 in nature and (2) delivery in a face-to-face and online mode. Our journey in creating a graduate-level information systems (MSIS) program for face-to-face and online delivery using the Backward design approach led to myriad learning opportunities that we identify and discuss in this paper. What resulted was an integrated curriculum with interrelated courses—not just a collection of courses grouped together and called a program. Although the context may differ (the program type, student body, delivery mode, and so forth), the lessons learned and the process described can provide insights that can be adapted by other institutions seeking to use the backward approach to program design.

REFERENCES


Please let us know, if you need the entire reference list. We did not include eight references in the reference list due to space constraints.


