

## RUBRIC FOR ASSESSING OUTCOMES IN BUSINESS INFORMATION SYSTEMS VIA CAPSTONE PROJECTS

Bonita M. McVey, St. Norbert College, [bonnie.mcvey@snc.edu](mailto:bonnie.mcvey@snc.edu)  
Kathleen K. Molnar, St. Norbert College, [kathy.molnar@snc.edu](mailto:kathy.molnar@snc.edu)  
David C. Pankratz, St. Norbert College, [dave.pankratz@snc.edu](mailto:dave.pankratz@snc.edu)

---

### ABSTRACT

*In recent years, assessment has become increasingly important in education. Additional focus has been placed on assessment by accreditation agencies with the result that all disciplines must participate in assessment. A central component of the assessment plan that we developed for assessing outcomes of our Business Information Systems concentration within the Computer Science major at St. Norbert College takes advantage of our capstone course and the projects that our seniors complete as part of the course. The rubric we use to assess outcomes, its development and evolution, and its effectiveness are presented in this paper.*

**Keywords:** Assessment, Capstone, Rubrics, CIS Program

### INTRODUCTION

In recent years, assessment has become increasingly important in education. Additional focus has been placed on assessment by accreditation agencies with the result that all disciplines must participate in assessment. A central component of the assessment plan that we developed for our Business Information Systems (BIS) concentration within the Computer Science major (CS) at St. Norbert College (SNC) takes advantage of our capstone course in which seniors are assigned 'real world' projects. As part of our assessment plan, we developed a rubric for assessing outcomes using the senior projects. The rubric we use to assess outcomes, its development and evolution, and its effectiveness are presented in this paper.

St. Norbert College is a small liberal arts college of approximately 2000 students. In spring 2005, the Computer Information Systems (CIS) and CS majors were combined into one major (CS) and from that three separate transcriptable concentrations were created: computer science (CS), business information systems (BIS) and graphics design and implementation (GDI) [2, 7]. Regardless of track, all CS majors complete a common core of five four-credit courses, the capstone course, and a comprehensive project. To

this point, the BIS track majors complete a group project instead of an individual project.

### OUR ASSESSMENT PLAN

The process of developing a firm assessment plan for the CIS major began in 2002. As suggested by a 2002 forum on Information Systems curricula trends [6], a CIS program should complement technical skills with business competency and emphasize the dual role of the business generalist with the information technology specialist. Specifically, the CIS program prepares students to effectively function as part of a project team in the analysis, design, development and implementation of a computer information system. The CIS program therefore built on two major academic components: Business Administration and Computer Science with members from both disciplines participating in instruction and assessment. Our initial assessment plan focused on the functional areas of business administration, CIS and the system project with achievement demonstrated by the Major Field Achievement Test (MFAT) in Business, standard exam questions in specified computer science required courses and evaluation of the system project, respectively. However, after the combination of the majors, a reassessment of the plan was undertaken.

Ultimately, the Computer Science discipline is responsible for the assessment of the CS major. The original assessment plan for CS, before the addition of the BIS and GDI concentrations, was designed and implemented one year after that of the CIS major, and the plans differed significantly in their approaches. However, both made significant use of the capstone courses, CS 460 and BA 445, along with senior projects. The assessment process showed that the goals and objectives for the CS and CIS majors were nearly identical for the courses that were part of the common core. It also showed that with respect to capstone courses and projects, the primary difference in goals and outcomes was in the area in which students applied core knowledge. We discovered that we shared the same goals and outcomes for our students, examined methodologies for assessing them, and combined the capstone courses into one.

## Objectives of the CS Major

Since CIS now falls under the CS major as the BIS track, our objectives started with the learning objectives proposed in 1999 when the original CS major was created. Additionally, Computing Curricula 2001 (CC2001) [4] had been recently released by the ACM and it also recommended objectives. Finally, we also examined Information Systems curricula trends [6]. As a result of examining these three documents, we identified the following objectives for all CS majors at SNC:

1. demonstrate a reasonable understanding of the subject areas and fundamental concepts.
2. demonstrate adequate competence in the methodologies of computer science, experimentation, and engineering.
3. understand the historical, social, and ethical context of computer science and the profession.
4. be able to identify/integrate/apply knowledge to design and build a system of reasonable complexity that solves a loosely specified problem.
5. be able to cope with change in computer science.
6. demonstrate self-management and team management.
7. demonstrate effective oral and written communication.

## Methodologies

It seems natural to assess, at least in part, all the above objectives in the capstone course. Objectives 1, 2, 4-7 are particularly assessable in the project component of the capstone course. However, we decided that it was important to assess outcomes at the course level as well. Therefore, objectives 1-3 are assessed throughout the curriculum. While researching various assessment methodologies, we discovered that Blandford and Hwang [3] and Fitzgerald, et al., [5] provided several ideas for assessing outcomes of specific courses. We have incorporated one of these into our assessment plan primarily using exam questions, lab assignments and homework assignments aimed at specific course outcomes in certain CS courses and in the BA course that emphasizes system analysis and design.

In Fitzgerald, et al., [5] and Sanders and McCartney [8], we found methodologies used to assess outcomes at the completion of the major. We decided against using senior exit surveys, alumni surveys and employer surveys for many of the same reasons given in Sanders and McCartney [8]: although information was being collected at SNC by the Alumni Office and the Office of Institutional Effectiveness, the few

number of responses would not provide reliable data. We also decided against a written comprehensive exam for the time being but did not rule out giving one in the capstone course in the future. Learning from the CIS assessment plan that used the MFAT and those responding in Sanders and McCartney [8], we know that some students do not take the exam seriously since scores are unavailable until after course grades are assigned. Ultimately, we chose to combine two methodologies from Sanders and McCartney [8] to create an assessment instrument, a 'portfolio' that consists solely of the capstone project and an 'oral exam' that consists of the three presentations required during the capstone project. All that was left was to adapt the rubric initially developed for the single track CS major to assess outcomes in all three tracks of the CS major via the capstone project.

In the past the CIS project grade was the grade given to all students on that group project unless there were peer evaluation discrepancies. With assessment we found we needed to also determine an individual grade for every CS major with a BIS concentration. Therefore, the BIS students are required to fulfill the CS requirements of a portfolio and an oral exam, both as a group and individually.

## THE RUBRIC

The rubric we developed for assessing objectives via the project component of the capstone course, which appears in Table 1 at the end of this paper, is the focus of this paper. Before examining the rubric, it is essential to examine the BIS capstone project. It can be summarized as follows:

- (a) Each CS major with a BIS concentration is assigned a 'real world' team project to complete, as is traditional in computer information systems. The faculty determines the project and normally the clients are outside of the SNC community.
- (b) Each CS major with a BIS concentration gives a group walk-through of their project near the middle of the term (approx. 30 minutes), a public team demonstration of their project two weeks before the end of the semester (approx. 45 minutes), and a group **and** individual final defense of their project to the CS faculty during the last week of the semester (approx. 45 minutes).
- (c) Each CS major with a BIS concentration is required to maintain a website that contains their journal, resume, CS philosophy statement, commentary papers, and their entire project. The journal should be updated weekly and address project description and requirements, progress, methodology, experiments, research, and

decisions. Members of a project team maintain individual websites as well as a project website. Over time, projects have covered a wide variety of areas including: Faculty nominations and voting system, the DePere Community Center Project, DePere Building permit project, and the Pulaski Community Middle School Parent Network database.

### Adapting the Rubric

After discussing our objectives and our capstone projects, a rubric seemed to be the right tool for measuring these outcomes. (Arter and McTighe [1] provide valuable information for developing and using rubrics.) Adapting the rubric was difficult for we needed it to be generic enough to apply to a wide variety of projects yet specific enough to obtain meaningful results. It truly helped to have had previous experience with examining capstone projects. Table 1 shows the resulting set of categories in our rubric.

We chose to use a 4-point Likert scale for scoring. In this scale a 4 is exceptional, a 3 signifies that a student has met our expectations for an SNC computer science major, a 2 means they fell short of our expectations but has some understanding, and a 1 signifies that the student has little or no understanding. We knew that using either a 3-point or 5-point scale would allow us to choose the easy way out by assigning the middle value but also that the 3-point scale would not provide enough information.

### Evolution of the Rubric

When adapting the rubric to assess outcomes for the three concentrations of the CS major, we made a few changes. Some changes were to the rubric itself and other changes were in the process of using the rubric.

Initially, there was one category for **Documentation**. As a result of combining CIS with CS, we were missing an important assessment component, external documentation. Therefore, we split the **Documentation** category into **External Documentation** and **Internal Documentation**. Particularly for BIS students, external documentation is an essential component of any systems project. External documentation includes everything from user manuals to installation guides, system service requests and system services requirements to database documentation and administrative documents. Although the focus in CS is typically on internal documentation, this additional emphasis on external documentation has proved to be beneficial to all CS majors.

A second significant change to the rubric was with the category now called **Robust/Useable**. Originally the category was titled **Robust** and referred solely to the reliability of the solution and its implementation. Again, a missing component for the BIS projects was an evaluation of the user interface, which directly impacts the usability of any project. For example, the initial faculty nomination and voting system was reliable software that faculty found difficult to use. A following BIS capstone project was primarily to revamp the user interface of this system to be more user-friendly. Once again, largely due to the usability requirement of BIS projects, the user interface has become a point of emphasis in all CS capstone projects. Given the evolving nature of computer science, we will continue to refine this category as well as the rest, as our rubric evolves.

Initially, CS majors gave a walk-through of their project in March and an open demonstration to faculty, students and guests at the end of the semester. The public demonstration did not support questions by the faculty necessary to properly assess the project. Therefore, seniors are now required to give the walk-through, a public demonstration, and a final defense attended only by faculty. The completed project must be provided to the faculty one week before the defense. For students completing the BIS concentration of the CS major, BA faculty with expertise in Information Systems join the CS faculty and attend the defenses. Following the defenses, these faculty members meet and complete the rubric for each BIS project, with the faculty member with expertise in IS leading the discussion of the categories such as **Identify, Integrate, Apply Knowledge, Robust/Useable** and **External Documentation**.

Using the rubric scores to assess whether or not our students meet the desired outcomes means doing some analysis of the scores. For each rubric category, we simply compute the percentage of students who meet or exceed the score of 3 (recall that a score of 3 means the student met our expectations for SNC computer science majors). Next, compare that percentage to a benchmark that would signify that we are reaching our objectives. We chose a modest benchmark of 75%; thus, if the percentage of students scoring a 3 or better is below 75%, we will address that area. Using this method, problem areas have been identified. Additionally, the rubric should not only assist us in identifying areas needing improvement but also in verifying that changes we are making are truly helping our students.

## THE FUTURE

Although we have limited experience using the rubric to assess the learning outcomes of students completing a group project, we have decided to make some changes. The rubric was initially designed to assess the outcomes of individuals and has proven effective when used in this way. However, we found it cumbersome when attempting to assess the outcomes achieved by members of group projects due to the limited participation of some members in various aspects of the project.

As a result, the BA and CS faculty have decided to make the following changes. All CS majors, including those completing the BIS concentration, will be assigned a suitable individual capstone project. In this way we hope to better assess the learning outcomes of all CS majors. However, participation in a team experience, always a part of computer information systems programs, has now become a point of emphasis in CS programs. Therefore, a team experience will now be required of all CS majors, not just the BIS students. A team project is now a requirement in all upper-level application courses and team performance will be assessed at that time. The CS faculty sees this as an improvement to all concentrations within the major.

Although we began the development and implementation of our assessment plan somewhat begrudgingly, we have become true believers in assessment. The rubric used to assess outcomes via capstone projects has proved to be incredibly valuable in helping us improve our CS program. In addition the synergy created by the collaborative efforts of the CS and BA/IS faculty in this process has not only created a robust rubric for assessment, but has greatly enhanced our CS major program.

## REFERENCES

- [1] Arter, J. and McTighe, J. Scoring rubric in the classroom: using performance criteria for assessing and improving student performance. Corwin Press. 2001.
- [2] Blahnik, J., McVey, B. and Pankratz, D. Adding concentrations to the cs major: our dean calls us 'innovative'. In *Proceedings of the Thirty-seventh SIGCSE Technical Symposium on Computer Science Education* (Houston, 2006) ACM Press, pp. 191-194.
- [3] Blandford, D. and Hwang, D. Five easy but effective assessment methods. In *Proceedings of the Thirty-fourth SIGCSE Technical Symposium on Computer Science Education* (Reno, Nevada 2003) ACM Press, pp. 41-44.
- [4] CC2001 Joint Task Force. (Dec. 2001). Computing Curricula 2001: Computer Science. ACM/IEEE-CS. Online
- [5] Fitzgerald, S., Fienup, M., Kubilus, N. J., Lopez, D., and McDonald, M. Student outcomes assessment: what works and what doesn't. In *Proceedings of the Thirty-first SIGCSE Technical Symposium on Computer Science Education* (New York, 2000) ACM Press, pp. 405-406.
- [6] Gorgone, J., Davis, G., Valacich, J. Topi, H., Feinstein, D. and Longenecker, H., *IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems*, <http://www.aisnet.org/Curriculum/index.htm>.
- [7] Molnar, K., McVey, B. and Pankratz, D. Collaborative Teaching: Changing the Way We Educate Computer Information System Majors in *Issues in Information Systems*, Volume VII (1), 77-82, 2006
- [8] Sanders, K. and McCartney, R. Program assessment tools in computer science: a report from the trenches. In *Proceedings of the Thirty-fourth SIGCSE Technical Symposium on Computer Science Education* (Reno, Nevada 2003) ACM Press, pp. 31-35.

Table 1.

Computer Science Capstone Project Assessment Rubric							
	Solves the Problem	Effort, Motivation	Identify, Integrate, and Apply Knowledge	Adaptability	Self Management	Oral Communication	System Dev. Life Cycle
4	And more!	Awesome	Optimal selection of data structs, algs, tools, hardware; Creative adaptation of data structs, algs, tools, hardware	Systematically researches, designs, makes progress, and masters the new stimuli	Keeps instructor informed; Forward thinking/planning; Accelerated progress; Thinks of extensions	Exceptional	Adapted formal methodologies to specific project components
3	Fulfills requirements	Expected	Integrates concepts from CS courses; Uses effective judgment in selection of data structures, algorithms, tools, hardware	Relates to similar concepts; Does initial research on unfamiliar concepts; Explores and experiments with new stimuli	Little guidance; Independent; Knows {when, where, who, what} to ask; Manages timeline; Steady progress	Proficient in tech. language, Approp. media, Addresses varied audience, Approp. examples, Responds to questions	Uses approp. methodology; Executes all steps approp; Develops a plan
2	Not completely	Half-hearted, Minimal	Simplistic use of data structures, algorithms, tools, hardware; Limited integration of CS concepts	"Don't know where to start" "I am lost" Initially needs to be told where to look for information; Unable to identify similar concepts	Considerable guidance; Handed information; Must be given milestones (assignments); Doesn't seek guidance when necessary	Adequate	Has trouble w/ steps; Incomplete analysis; Incomplete testing;
1	Minimal progress	Inadequate	No integration of CS concepts; Doesn't correlate CS concepts to project	"Tell me what next"	Parasite; Unable to progress independently	Inadequate	Skips steps; Unable to analyze; Doesn't understand design;

Computer Science Capstone Project Assessment Rubric (continued)							
	Software Engineering Principles						
	Robust/Useable	Portable	Extendable	Efficient	Modular	External Doc	Internal Doc
4	Fully reliable; Helpful, consistent, "production quality" interface	Fully	Fully	Fully	Fully; libraries, dlls, drivers	Fully, exceptional	Fully
3	Mostly reliable; Understandable interface	Mostly	Mostly	Mostly	Mostly	Proficient, readable, clear, succinct; complete; useable; organized website; referenced;	Mostly
2	Minimally reliable; Inadequate interface	Minimally	Minimally	Minimally	Minimally	Incomplete instruction on use; Not easy to find materials	Minimally
1	Unreliable; Confusing interface	None	None	None	None	Incomplete; Unorganized;	None