7-2013

Using case studies to design and deliver technology-centered computing education courses: An innovative approach from an undergraduate Information Systems program in Singapore

Ilse BAUMGARTNER
Singapore Management University, ibaumgartner@smu.edu.sg

Using Case Studies to Design and Deliver Technology-centered Computing Education Courses: An Innovative Approach from an Undergraduate Information Systems Program in Singapore

Ilse Baumgartner
Singapore Management University
80 Stamford Road
Singapore 178902
+ 65 6828 1917
ibaumgartner@smu.edu.sg

ABSTRACT

While the advantages of using case studies as an educational vehicle in computing education appear to be more than obvious, there is a very limited amount of research works or practice papers reporting on actual implementations of undergraduate or graduate computing courses which would be largely based on case studies. This conference contribution reports on selected best practices of course design and delivery implemented in one of the core courses of the Bachelor of Science (Information Systems Management) degree program (BSc (ISM)) offered by the School of Information Systems (SIS) at the Singapore Management University (SMU). Nearly all assessments, exercises, in-class activities and lectures of this selected course (Enterprise Web Solutions) are centered around one specific case study which has been written specifically for this course.

The paper reports on the practices implemented in this course to design and deliver all hands-on components of the course, it discusses the approach used to tie the lecture component of the course to the case study, and it briefly discusses the rationale of using a fictitious company for the case study of the course (instead of using a real company and real scenario).

The paper concludes with a series of recommendations to course designers and educators in the computing education field who are involved in the design and delivery of technology-centered courses and who are interested in exploring the use of case studies in their courses. While the paper is describing best practices implemented within the frame of an Information Systems program, these practices are largely applicable to any other computing education field: computer science, computer engineering, software engineering, information technology, or informatics.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education – *curriculum, information systems education.*

ITiCSE'13, July 1–3, 2013, Canterbury, England, UK.

General Terms: Design.

Keywords: Course design, case studies, computing education, undergraduate information systems curriculum.

1. INTRODUCTION

This conference contribution reports on selected best practices of course design and delivery implemented in one of the core courses of the Bachelor of Science (Information Systems Management) degree program (BSc (ISM)) offered by the School of Information Systems (SIS) at the Singapore Management University (SMU).

The paper unfolds the following manner.

In chapter two, the author undertakes a brief review of the case study methodology in the context of university-level computing education.

Chapter three introduces the reader to the Enterprise Web Solutions course, a senior undergraduate core course which has been used to demonstrate the case-based course design and delivery approach described in this paper.

In chapter four, the author describes the case-based approach used in the Enterprise Web Solution course in detail. This chapter describes the rationale of using a "self-written" case for the course instead of utilising an existing commercially available case; moreover, this chapter also briefly describes the reasons for using a fictitious company instead of a "real" one for the case study of the course. Further, this chapter also offers detailed insights into the case-based approach applied to deliver this course. Particular attention is devoted to the design and delivery of the laboratory components as well as the final project component of the course and the role the case study plays for those two assessment components.

In chapter five, the author briefly discusses selected recommendations for computing educators interested in case-based course design and delivery.

The paper concludes with a brief summary particularly emphasising suitability of the best practices described in this paper for any computing education field: computer science, computer engineering, software engineering, information technology, information systems or informatics.

2. CASE STUDIES AS A PEDAGOGICAL METHOD IN COMPUTING EDUCATION

2.1 Introduction

According to Lawrence a useful case study is "the vehicle by which a chunk of reality is brought into the classroom to be worked over by the class and the instructor. A good case keeps the class discussion grounded upon some of the stubborn facts that must be faced in real life situations." [1]

Case studies have been used in higher education fields such as medicine, business or law for decades [2]. Case studies are commonly used to expose students to problem solving approaches which cannot be meaningfully communicated using standard textual descriptions [3]. Case studies are usually written in a common language and are based on an easy-to-comprehend story, thus, they usually leave a more lasting impression on students than standard higher education text books written in a technical or academic jargon. Moreover, the practicality and vividness of information, data and artefacts embedded into the case study create additional aids for students and make the case study far easier "digestible" for students than a standard academic text.

2.2 Case Studies in the Computing Education Field

While some progress has certainly be achieved, it appears that the majority of computing education still happens with the frame of traditional lecture-based classes [4]. The main reason for this might be that computer science, information technology or information system courses are taught primarily by people who have advanced university degrees in one of those fields, but little exposure to practical work in real industry settings. The knowledge gained through the university-level computing education remains passive and disconnected from the real-world environment. Consequently, it is not easy for the graduating students to apply this knowledge to actual problem cases.

Nevertheless, in recent years, new approaches have been introduced in computing education – such as project-based learning [5], problem-based learning [6], situated learning [7] or inquiry-based learning. Problem-based learning and project-based learning approaches are rather similar (both approaches are targeted at successfully completing a defined task). However, project-based learning appears to be a more structured approach in the sense that the goal of the project seems to be more explicitly defined and there is more direct guidance from the instructor available. Contrary to that, the problem-based approach focuses on the learner's role in identifying outcomes and parameters for success [8]. The situated learning approach has been adopted from the social sciences field, being primarily based on constructivism.

Case-based learning seems to be a synthesis of different approaches, such as problem-based learning, project-based learning or inquiry-based learning. Moreover, case-based learning model can be considered an answer to the current challenges of computing education programmes to produce "technology-savvy" as well as "business-literate" graduates who are well prepared for the work in the industry under the current ever-changing conditions.

Different perspectives on the same phenomenon frequently create different problems. In the conventional problem-based or project-based learning models, students deal with problems through project work, they create solutions which address specific given problems. However, this results from the perspective that the given problem is stable and will never change. In case-based scenarios, where students are forced to evaluate the given situation from different perspectives, problems are no more stable, situations can change, perspectives can change, accordingly, solutions can change. And, consequently, the solution produced cannot be anchored in one, single, stable problem anymore.

In case-based learning, the instruction is largely anchored in cases which create intense and realistic working scenarios and which are very suitable to create motivating and well-aligned learning environments.

3. ENTERPRISE WEB SOLUTIONS COURSE

The Enterprise Web Solutions (EWS) course is a third year core course focusing on the design, implementation and governance of an enterprise portal.

Using an enterprise portal as an example the students are introduced to the full lifecycle of a complex enterprise-wide application. An enterprise portal (like many enterprise-level implementations) has the potential of deeply transforming the key business processes of a company, it can break down the communication barriers in a company and increase its operational efficiency, an enterprise portal is capable of transforming the way of how an organization is communicating with its external partners, suppliers, customers. However, despite the comprehensive and powerful enterprise portal systems currently available on the market the implementation of an enterprise portal is frequently ending in a disaster. The main reason for it is the simple assumption that the value of an enterprise portal is provided through the enterprise portal technology per se.

Using a "real-world-like" case, the course shows the students that an enterprise portal project success is rather driven by having a clear vision, objective, understanding of end-users and their needs and understanding the underlying business processes and operations in a company.

The EWS course is delivered in both terms of the academic year – while in term 1 (August to December) the number of students enrolling in the course is approximately 160, term 2 has the approximate enrolment number of 120 students. Due to the small class sizes at the SMU School of Information Systems (the maximum number of students per section or class is 40), the first term has usually 4 sections of the course, and the second term 3 sections of the course.

A typical 3 hours session of the EWS course is divided into two main parts: the theory part and the practical part. While during the theory part, the students are introduced to numerous portal-related concepts, techniques and technologies, the practical part is a purely hands-on based session where the students perform numerous lab exercises, project work and in-class exercises. Moreover, the course also has a large final project asking the students to build a large multi-tier portal-based system.

While the EWS course can be considered a technology-centered course, it has an extensive business and process management focus, too. Figure 1 shows the general structure of the EWS course:

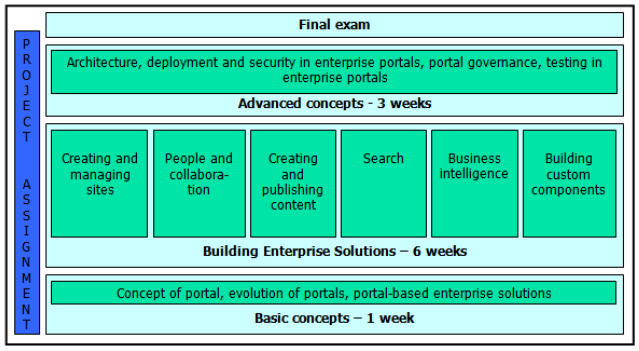

Figure 1: Structure of the Enterprise Web Solutions course

The EWS course uses a comprehensive package of Microsoft technologies. Currently, the students use Microsoft SharePoint Server 2010 as the principal tool for building the enterprise portal infrastructure, Microsoft Visual Studio 2010 Ultimate is used for custom component design and development, Microsoft SQL Server 2008 Professional serves as the database server, and Internet Information Services 7.5 is used as the web server. Additional software used in the course is Microsoft Search Server 2010, Microsoft Office 2010, Microsoft Expression Web and Microsoft SharePoint 2010 Designer.

4. CASE STUDIES IN THE ENTERPRISE WEB SOLUTIONS COURSE

4.1 Introduction

The Enterprise Web Solution course has been designed with the principle aim of providing the students with a holistic and comprehensive view of a large scale, multi-tier enterprise-level system implementation.

Currently, many companies attempt to expand and to enhance their organisational portals to systems which act as a "single-point-of-entry" to most organizational systems and processes. More and more, the enterprise portals are built up to a framework which integrates content management, business processes, collaboration and information sharing, business intelligence and many internal as well as external systems, applications and services. Moreover, enterprise portals are built on open architectures and open APIs which enable integration with numerous backend systems, legacy applications and tools used in an organisation.

Consequently, implementation scenario of an enterprise portal is complex and multi-sided, it involves intensive collaboration of numerous parties in an organization (top management, leaderships of individual departments or functional units, key business users in all departments or functional units, IT personnel) and it aims at building a system which is not only capable of addressing small, isolated business problems in individual departments or functional units in an organisation but which also truly connects the entire organization.

Considering this, implementation case of an enterprise portal represents a highly suitable setting for a course which not only wants the students to develop technical capabilities but which also wants the students to considerably enhance their "business literacy". Moreover, the implementation scenario of an enterprise portal is highly suitable for a course which is looking for an efficient and meaningful way of integrating individual student work into a project which is performed by students as a group (the individual and the group work, however, still being clearly distinguishable and assessable).

4.2 Use of Case Studies: the Rationale

Already during the initial design phase of the course, the decision was made to make this course as practice-oriented as possible. While the students certainly need to be exposed to numerous theories and concepts, the main focus here needs to be hands-on work and experience in building a real enterprise portal for a real company.

As already discussed in the literature review, the use of case studies in the computing education is becoming more and more popular as the case study methodology is particularly apt to expose students to practice-relevant scenarios and to impart practice-relevant skills and knowledge. Thus, considering the nature and the aims of the EWS course, the use of case studies in this course seemed to be a very promising way of proceeding.

The first problem here, however, was a real lack of suitable **technical** case studies. While there are countless business-centered case studies, there are hardly any case studies available incorporating the level of technical detail which is needed in order to be suitable for a technology-centered course such as the EWS course. This lack of suitable case studies led to the decision of creating a technical case study "in-house" to be used exclusively for this course.

Next major question which had to be answered was the decision on the number of cases to be used for the course. How many case studies are needed for this course? Should in-class activities and exercises (e.g., laboratory components, in-class design exercises etc.) be based on a different case study than the final project of the course? Should the case studies used to illustrate theories and concepts during the lecture session be different from these case studies used in the practical sessions?

After having carefully considered the nature of the course and the aims of the course, the decision was made to build the **entire** course experience around **one** particular case. The main rationale here is the following: an implementation scenario of an enterprise portal (similarly to any large-scale organisation-wide system) represents *per se* a complex and multi-layered venture. To successfully understand at least the basics of this implementation undertaking, the students need to be exposed to many perspectives in such a project – for example, the perspectives of the top management, IT personnel, or key business users. The students need to deal with different business problems in different organisational units – and seek for their solutions. While many of those issues can be resolved in an isolated manner, many of them are directly linked to numerous other organisational concerns. Thus, the students need to understand the causalities and interdependencies of those problems – and they need to appreciate how interconnected business operations and business processes are in an organisation. Based on this understanding, the students are expected to build an integrated and cohesive solution which is supposed to effectively address the identified issues and problems of the case study company.

Consequently, using only one comprehensive case study for the entire course seemed to be the most effective way to proceed.

An additional, though program-internal but not less important reason for using a case study for the EWS course was learning outcomes and competencies framework established and implemented within the BSc (ISM) program at the SMU School

of Information Systems [9]. This framework consists of program-wide learning outcomes shared across all courses of the program (representing a nested three-level construct) and course-internal competencies which address the learning outcomes. The EWS course defines 37 course-specific competencies, figure 2 displays selected competencies of the course:

2. IT architecture, design and development skills

2.2. Software and IT Architecture Analysis and Design skills

***Competency 1**: Document server architecture for an enterprise portal based on a given business case.*

***Competency 2**: Explain the basic steps of an enterprise portal planning process.*

***Competency 3**: Explain the most important considerations which need to be taken into account when developing search requirements for the enterprise portal.*

***Competency 4**: List the most important end-user facing search features in enterprise portals.*

***Competency 5**: Create a portal topology document for an enterprise portal based on a given business case.*

***Competency 6**: Perform basic user needs analysis and document the outcomes of this analysis when designing and implementing an enterprise portal.*

Figure 2: EWS course competencies (excerpt)

The competencies defined for each course represent observable and measurable skills which all students are supposed to possess when graduating from a specific course. The Learning Outcomes and Competencies Framework does, thus, represent an essential reference system for all students of the program. Consequently, the students are particularly looking for assessments or activities where they are able to develop and to clearly demonstrate the attainment of the competencies that are specified for this course. Moreover, the students are also particularly interested in the evidence that the competencies which they have attained are valued and applicable to practical settings in the industry.

Case-based learning and case-centered course is particularly apt to provide such evidence and to strengthen the students' self-confidence in being effectively prepared for work after the completion of the studies.

4.3 Use of the Case Studies: the Approach

The case written for this course is used for nearly all assessments, all in-class exercises and all in-class discussions of the course. Figure 3 summarises the use of the case study across all assessments and activities of the course:

EWS Course Assessments

	Assessments and activities	Use of the case study
1	Laboratory exercises	✓
2	Quizzes	–
3	Midterm assignment	✓
4	Capstone project	✓
5	Final exam	✓
6	In-class exercises and discussions	✓

✓ Extensive use

✓ Partial use

– No use

Figure 3: EWS course assessments activities and their use of the course case study

Due to the reasons discussed above, this case was written exclusively for this course. Moreover, this case is based on a fictitious company. The reasons for not using a real company are obvious.

Firstly, it is essential to keep in mind that the course in this context is not "serving" the case, but the case is "serving" the course. Which means, for the approach to succeed it is absolutely essential to design a case study which perfectly fits into the projected course structure, which clearly incorporates the required level of detail, which serves not only as a basis for the capstone project of the course but which is also suitable for the laboratory exercises and in-class design exercises. Next, the case has also to incorporate suitable examples to illustrate the concepts and theories covered during the lecture sessions, and it also has to provide sufficient basis for in-class discussions. Designing a "course-proprietary" case study based on a fictitious scenario is the only reasonable way to achieve those goals.

As already mentioned, the case is used in the laboratory components and in the final project of the course. The approach of using the case in those two components is considerably differing from a standard use of case studies in university-level computing education courses. The following section will explain this difference in detail.

The case consists of two major parts – the "base-line story" and the "mini-stories".

The "base-line story" is approximately 12 pages long consisting of textual material as well as appendices and graphs explaining technical and organisational aspects of the company and the case. The "base-line story" introduces the students to the industrial setting of the case company, the history and organisational structure of the company, and it provides very detailed information on the organisational business processes and IT infrastructure of the company. Moreover, the case contains extensive information on numerous issues, problems and challenges which the company is currently experiencing and which the company is hoping to address through the implementation of an enterprise portal. Those issues are equally organisationally and technically oriented.

The "mini-stories" are around half page long and these "mini-stories" principally focus on selected isolated problems and issues in the company which might be addressable using selected portal-based tools or techniques. These "mini-stories" complement the "base-line story" through providing additional information on the case company and explaining specific issues more in-depth.

While the "base-line story" of the case is principally used for the capstone project (and partially also for the in-class discussions and in-class exercises), the "mini-stories" are mainly used for the laboratory components.

For each of the laboratory components (a laboratory component occupies approximately half of the entire 3 hours session), the students receive one "mini-story" upon which the entire laboratory material is built. Each laboratory component consists of two major parts: the guided part and the problem exercise part. In the guided part, the EWS course is using a video-based approach to teach students techniques, concepts and technologies which enterprise portals expose and which might be applicable to the given situation. After having read the corresponding "mini-story", the students watch the videos of the specific laboratory component, they learn how to use the techniques and tools taught

through those videos, and they attempt to link the tools and techniques to the relevant "mini-story" and to the issues and problems of the organisation described in this "mini-story". Once the guided part has been completed, the students attempt the problem exercise part – a part which is asking the students to refer to selected problems described in the "mini-story" and to propose and implement a solution of those problems (this part the students need to complete independently and without any guidance, and it might require some additional independent research).

In summary, every week, the students are building upon the previous laboratories – i.e., step by step extending their solution through additional components. After 12 laboratory sessions, the students have a basic complete end-to-end enterprise portal implementation based on the information given in the 12 "mini-stories".

As far as the capstone project of the course is concerned, the students need to consider information given in both components of the case study – while the "base-line story" is supposed to serve as the main basis for the project, the students also need to consider all the information given in the 12 "mini-stories". The final project is principally concerned with building a complete end-to-end portal-based solution for the case company – including designing the topological structure of the proposed solution, the server architecture of the solution, building the actual portal implementation and proposing a governance plan for the solution.

While a specific part of the project needs to be done in a group, the project also incorporates a large individual component where the students need to build individual web components and integrate them into the common group solution. For this, the student project group jointly analyses the given case study and identifies small problems or issues in the case company which can be solved by individually built web components, those identified issues then are assigned to individual project members for solution and implementation. Subsequently, the created small components are integrated into the main portal solution which students build as a project group.

The theory part of the course also extensively uses the course case and attempts to link the problems highlighted in this case to the underlying concepts, theories and techniques. Or – in other words – the theory part, just like the hands-on part, is fully centred around the case of the course and constantly refers to this case when teaching the students certain concepts or underlying meanings.

All in-class exercises (e.g., conceptualising and designing a workflow, drafting navigation for the portal implementation, designing visual layout of the portal pages etc.) are also built on the course case study – and are principally used to prepare the students for similar tasks to be executed for their capstone project implementation.

Lastly, the final exam component is also fully based on the course case study. While the laboratory exercises and the capstone project are comprehensive hands-on components, the final exam tests the students' understanding of the underlying theories and it tests the students' capabilities to apply those theories and concepts to a specific case.

Thus, linking all course content – laboratory exercises, in-class tasks, lecture material and the capstone project – to a given real-world-like scenario creates a holistic and fully integrated learning and teaching experience in the course, transparently connecting the underlying theories and concepts with practical applications, and, thus, educating technically competent, but also "business-savvy" IT professionals.

5. DISCUSSION AND RECOMMENDATIONS

The EWS course has been using cases for eight subsequent terms (i.e., for four subsequent years). The current approach in using one "real-world-like" case study serving as a basis for the entire course has been tested for almost three years. The following section will summarise some selected recommendations which have emerged during this time.

While there are many interesting aspects to report, the author of this paper would like to particularly focus on two aspects: the use of the 12 "mini-stories" for the laboratory exercises instead of using the main "base-line story" (i.e., following a "two-part" case study approach) and the use of the same case for nearly all assessment components of the course (including the final project of the course).

While the overhead of preparing and writing the case is definitely considerable when following the two-part approach, the experiences made in the course have shown that the advantages of this approach largely outweigh the additional work.

As indicated earlier, the laboratory work usually is conducted in the second half of the class, and the duration allocated to the laboratory exercises is around 1.5 hours. During this time, the students need to familiarise with the case situation for this particular laboratory session, they need to consider all the multimedia material provided for this session, and they need to read the laboratory instructions. All this takes considerable time. Using a large, continuous case study for such a purpose does not seem to be adequate as such a large case study might considerably distract students' attention and diminish their productiveness. Instead, small chunks of text describing small, rather isolated issues and problems have proven to be the most effective way of providing students with a limited but usable context for performing the laboratory tasks. These small chunks of text can be read quickly; the students can quickly understand the main points of this "mini-case" and can almost instantly start working on the actual solution. In addition, those small portions of text (which are provided to students one by one each laboratory session) are constantly referring to one and the same organisation, to same case participants, to one and the same industry this way allowing the students to subsequently build up a broader context and broader understanding of the underlying processes, operations and IT landscape.

While there might be arguments which speak against the use of the same case study for most of the assessment components of the course (particularly for the laboratory components and for the final project of the course), the experiences made in the EWS course show that this approach might actually considerably deepen the students' understanding of the topics covered in the course and it might very much strengthen students' confidence in being able to successfully participate in complex enterprise-level system implementations. As indicated earlier, the case written for this course is based on a fictitious company. This means, that – contrary to a "real" case which is anything but deliberately changeable and adaptable – the author of the case has the complete freedom in including all the details, data and aspects which are needed to fully cover the needs of all assessment

components of the course and all activities in the class. During the course, the students get exposed to different problems from very different perspectives, they have numerous opportunities in revising their initial perceptions of specific issues, they have the possibilities in re-creating and improving their proposed solutions, they have the opportunities of constantly deepening their understanding of the context and circumstances of the case – opportunities which the students would not have if the course used more than one case study or if the course used different case studies or different contexts for different course activities. The quality of the final project of the course has made this very obvious – as the capstone project of the course is implemented on the same case company as all the other assessment components and in-class activities are built upon, the students are able to implement solutions which are deep, which are technically very sound, which incorporate sophisticated conceptual approaches to organisational problems and which go beyond surface of the relevant case study.

6. CONCLUSION

This paper describes an innovative course design and delivery approach implemented in one of the core courses of the Bachelor of Science (Information Systems Management) degree program (BSc (ISM)) offered by the School of Information Systems (SIS) at the Singapore Management University (SMU). Nearly all assessments, exercises, in-class activities and lectures of the Enterprise Web Solution course are built around one specific case study which has been written specifically for this course.

The paper reports on selected practices implemented in this course to design and deliver all hands-on components of the course, it discusses the approach used to tie the lecture component of the course to the case study, and it briefly discusses the rationale of using a fictitious company for the case study of the course (instead of using a real company and real scenario).

While the paper is describing best practices implemented within the frame of an Information Systems program, these practices are equally applicable to any other computing education field: computer science, computer engineering, software engineering, information technology, or informatics. Thus, the author of this paper expect this contribution to be of interest to any educator or and course designer who is interested in exploring opportunities in using case studies to design and deliver technology-oriented university-level computing education courses.

7. REFERENCES

[1] Christensen, C.R., Teaching By the Case Method. 1981, Boston: Harvard Business School.

[2] Kreber, C., Learning Experientially through Case Studies? A Conceptual Analysis Teaching in Higher Education, 2001. 6(2): p. 217-228.

[3] Richards, L.G., et al., Promoting Active Learning with Cases and Instructional Modules. Journal of Engineering Education, 1995. 84(4): p. 375-381.

[4] Schilling, J. and R. Klamm, The difficult bridge between university and industry: a case study in computer science teaching. Assessment & Evaluation in Higher Education 2010. 35(4): p. 367–380.

[5] Coppit, D., Implementing large projects in software engineering courses. Computer Science Education, 2006. 16(1): p. 53-73.

[6] Nuutila, E., S. Törmä, and L. Malmi, PBL and computer programming: The seven steps method with adaptations. Computer Science Education, 2005. 15(2): p. 123-142.

[7] Cousin, G. and F. Deepwall, Designs for network learning: A communities of practice perspective. Studies in Higher Education, 2005. 30(1): p. 57–66.

[8] Savery, J., Overview of Problem-based Learning: Definitions and Distinctions. The Interdisciplinary Journal of Problem-based Learning, 2006. 1(2): p. 9-20.

[9] Baumgartner, I. and V. Shankararaman. Framing core and advanced competencies for undergraduate information systems program courses: Does the nature, level, complexity and audience of a course matter? . in 2012 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE). 2012. Hong Kong.