

Teaching Advanced Visual Basic .NET in a Distributed Classroom

MECHTILD JOHNSON

Computer Science Department

University of Wisconsin Washington-County

400 University Drive, West Bend, WI 53095

UNITED STATES OF AMERICA

http://washington.uwc.edu/about/faculty/johnson_m/default.html

Abstract: - A trend of declining enrollment in Computer Science (CS) is being felt at the 2-year University of Wisconsin (UW) Colleges, and it therefore does not seem an opportune time to expand the CS curriculum. Nevertheless, in 2003 the UW Colleges successfully introduced a new CS course in Advanced Visual Basic .NET, a course offered via distance education to the thirteen UW Colleges, which are geographically spread over the entire state of Wisconsin. Two issues bear examination: first, the success of this course, which is believed to be due to the distributed classroom model adopted for the course, and second, the interest of students in a course which is neither part of the standard CS curriculum, nor easily transferable to the 4-year UW Colleges.

Key-Words: - Distance education, Computer Science curriculum, Visual Basic .NET, Object oriented programming, Three-tier design, Web learning

1 Introduction

The thirteen 2-year University of Wisconsin (UW) Colleges serve students over the entire state of Wisconsin. The Computer Science (CS) program in the UW Colleges is designed to be easily transferable to the 4-year institutions within the UW System. The nationwide decline in CS enrollments reported in the 2003-2004 Taulbee Survey [1] has affected the UW System, and has been particularly hard on the 2-year UW Colleges, which are relatively small colleges, with a combined enrollment of 12,452 students [2]. Nevertheless, in 2003 the UW Colleges successfully introduced the new course Advanced Visual Basic .NET (Adv VB .NET), which is offered via distance education (DE). The successful offering of Adv VB .NET bears examining from two different perspectives that are believed to have contributed to its success: 1) The DE model adopted for offering the course, and 2) The interest in a course that is outside the standard CS1-CS2-CS3 curriculum.

2 The Distributed Classroom Model

Adv VB .NET is a 3-credit course, meeting twice a week for 75 minutes for 15 weeks. The instructor teaches from his/her office, sharing voice via telephone and sharing the computer screen via an Internet connection. The enrolled students meet in assigned rooms at their respective campuses, receiving sound and watching the computer screen. This model is generally referred to as a Distributed Classroom model [3]. In addition, course management software is used for teacher and students to

share resources on the web. The technologies used are now discussed in greater detail.

2.1 Instructional Communication

The communication used during the synchronous classroom time is managed by Instructional Communications Systems (ICS) [4], which is a support unit of the UW System. For the audio part of the classroom communication, ICS uses the teleconferencing call service WisLine, and for webconferencing ICS uses Wisline Web (WLW) [5], a technology based on the Microsoft Office Live Meeting software.

One of the nice features of WLW is that all sessions are archived, and students who miss a class can request the archived classroom session with instructor consent.

To the instructor herself, the use of this technology comes down to having a phone, a broadband Internet connection, and a browser. It is therefore just as easy for the instructor to teach from home as it is to teach from the office. This would not be possible without the infrastructure of an information technology (IT) support system within the UW Colleges.

2.2 IT Support System

Distance education has become part of the curriculum of all departments in the UW Colleges, and the necessary support staff is in place. Each of the thirteen colleges has IT staff dedicated to distance education. The IT staff makes sure that the DE room is open, that the computers are logged in to the correct website, that the telephone

connection is established, and, though the IT staff does not necessarily stay in the DE room, that both instructor and students are able to communicate with the IT staff instantaneously via e-mail or telephone should problems arise. In addition, student and instructor alike can reach the operators at WisLine Web if a communication problem occurs. If problems arise, the IT staff is also available to help out with the course management software, which is discussed next.

2.3 Course Management Software

The Course Management Software used by the UW System is Desire2Learn (D2L) [6]. D2L allows for asynchronous communication with the student. To explore all features of D2L is beyond the scope of this paper, and therefore only the key features the author herself uses are discussed here. The topics are discussed in the order in which they appear to the students on the toolbar shown in the Screenshot (Fig 1.): Homepage, Content, Classlist, Discussions, Dropbox, Quizzes and Links.

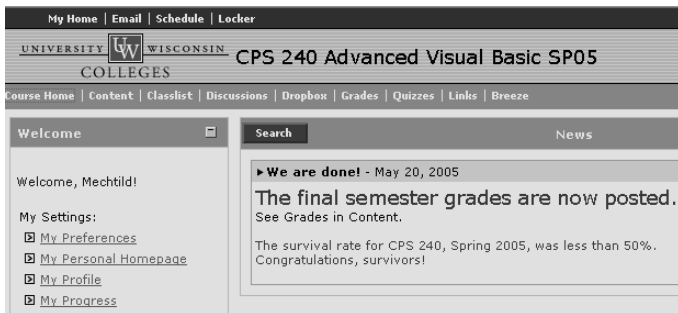


Fig 1: D2L Screenshot - Homepage

2.3.1 Homepage

The homepage of the D2L course section allows for up-to-date announcements, containing information such as when the next programming assignment is due, for what date the next test is scheduled, or what topic is currently under discussion.

2.3.2 Content Section

In the content section of D2L, the student can find postings of virtually all course materials used, which includes programming assignments, sample programs, and lecture notes. All PowerPoint presentations used during class time are also posted in this section. A frequent question that comes up during class time is: "Will this be posted in D2L?" A partial screenshot (Fig. 2) shows how the content section is broken up into modules.

2.3.3 Classlist

The Classlist is a primarily static look-up section which shows the names and email addresses of instructor and students. Icons indicate which course members are currently signed into D2L, and the names act as links to send paging messages to each other.

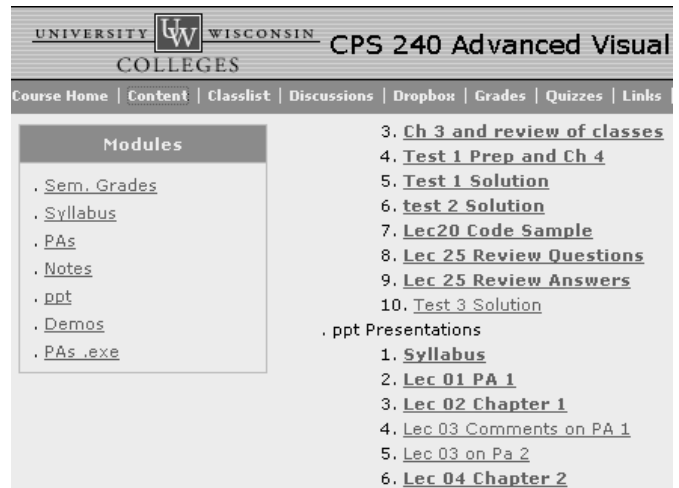


Fig 2: D2L Screenshot - Content

2.3.4 Paging

D2L has an internal instant messaging service referred to as "paging." The drawback of paging is that the person needs to be logged into D2L in order to be reached. In my own experience, I have found that the students prefer to use MSN Messenger for instant messaging to the instructor or among each other.

2.3.5 Discussions

In the discussion section, the instructor can set up forums for threaded discussions. Usually one forum is reserved for student interactions, while other forums require mandatory participation which is part of the active learning and grading process. Asynchronous discussions thereby become a valuable learning tool, as has been suggested by other authors as well [7].

2.3.6 Dropbox

The dropbox is used by the students to submit files to the instructor. Dropbox items can be linked to grading items, and the instructor has the option to post comments in the dropbox for the submitted work.

2.3.7 Grades

This feature allows students to look up their current grade standing.

2.3.8 Quizzes

From the many different quiz options I use mostly the multiple choice option to test the knowledge connected to the assigned reading. I have found automatic grading and recording of the grades to be a great convenience.

2.3.9 Links

At the beginning of the semester, this section contains barely more than an URL to the instructor's website and the URL to the textbook site, but during the semester, many URLs get added as students mention web resources of interest to the course.

3 The "Lecture" Format

Though the distributed class time, 75 minutes twice a week, is referred to as the "lecture" part of the course, I am reluctant to use the term "lecture" in a time where general pedagogy concentrates on active learning [8]. On the day of the lecture, the instructor typically calls in ten minutes ahead of lecture time (audio) and in WLW displays a slide of a virtual seating chart (video). These ten minutes are used by students to mark their presence on the seating chart, and to talk to the instructor informally. Within the 75 minutes of class time, I have primarily used the following four presentational techniques.

3.1 PowerPoint Presentations

Typically, the class time partly consists of introducing new course material via a lecture accompanied by a PowerPoint (ppt) slideshow, which is posted in D2L after class, so that students do not necessarily need to take notes. I do not use the ppt presentations offered by the textbook company, because besides my voice, I consider the slideshow the major means for compensating for the absence of face-to-face interaction between teacher and student; I, therefore, design my own personalized ppt presentations, enhancing them with illustrations and appropriate screen captures. During lecturing, the virtual seating chart is always nearby so that students can individually be called on to comment on the material presented. Student and instructor can mark the ppt slideshow in real time and add text to them as well; however, the text feature is relatively awkward, and what has worked better for written interaction during class time are the interactive text slides which are discussed next.

3.2 Interactive Text Slides

WisLine Web allows for adding interactive text slides in a ppt presentation before as well as during class time. The advantage of these text slides is that they easily

allow for student and teacher to write to the screen, as if a common Notepad were shared. This is a particularly advantageous feature for interactivity, and ppt slideshows can be interspersed ahead of time with interactive text slides. Though audio can be used to answer questions, certain questions such as asking the student for a programming code segment are better answered if the student can type the answer for the entire class to see. The voting feature discussed next serves a similar function, but has the advantage of involving all students at the same time.

3.3 Voting

WisLine Web also allows for adding voting slides in a ppt presentation before or during class time. On a voting slide, each student can click on an option and a bar chart displays the result in real time. Voting slides can be used to poll students on opinions, but more importantly, by asking students to select answers to multiple choice questions, voting slides can provide an excellent feedback to the teacher as to how well the students are following the presented material.

3.4 Application Sharing

Another feature offered by Wisline Web is application sharing. In the sharing mode, students can watch whatever application the instructor is running. In a programming course such as Adv VB .NET, application sharing allows the students to watch the instructor work within the Integrated Development Environment (IDE), assembling forms, and writing and debugging programs. Though the feature has many advantages, a caveat is in order here. Students watching the instructor program in real time is potentially more confusing than helpful, unless the instructor becomes very talkative, explaining each step, and doubling back repeatedly to explain where in the program the code is being written.

Regretfully, the student cannot interact directly with the application that the instructor is displaying. Though this feature of complete application sharing is technically part of WisLine Web, the feature as of now has been disabled because too often it has caused applications to hang.

At the end of the semester, students are asked to evaluate the distance education model used for the course, and generally the students express satisfaction with the model. The asynchronous sections of the course in particular, that is D2L, or communication via e-mail or MSN Messenger, prompts students to comment that they have the feeling that their distance education instructor is more accessible to them than some of their live instructors. Of course, the students are also asked to

evaluate the course itself, because no matter how good the distance education technology serves the student, the topics presented need to hold the students' interest for a course to be successful.

4 Adv VB .NET, Course Description

With declining CS enrollment, the UW Colleges have given priority for live presentation to CS courses which are part of the transferable CS core curriculum. The UW Colleges are currently re-adjusting the CS core curriculum to follow more precisely the objects-first, 3 semester sequence suggested by the 2001 ACM Curriculum Report [9]. CS 1 and CS 2 are offered in Java while CS 3 is offered in C++, thereby following a trend set by the 4-year UW colleges.

The offering of Adv VB .NET would not have been possible without distance education, because each of the 2-year Colleges is too small to fill CS courses outside of the transferable core curriculum. However, the success of the Adv VB .NET course cannot be solely attributed to distance education. The question therefore remains of why there are enough students willing to enroll in a course which is outside of the mainstream curriculum languages C++ and Java, a course whose transferability to the 4-year colleges is questionable. To examine this question let us look more closely at the Adv VB .NET course.

4.1 Prerequisites

The prerequisites for Adv VB .NET are basic algebra proficiency and previous programming experience (not necessarily in Visual Basic), or consent of instructor [10]. Many of the students enrolling in Adv VB .NET are Freshmen, and their backgrounds in programming vary greatly. The student is expected to be comfortable with basic fundamental programming constructs: variables, types, expressions, assignment statements, simple user I/O, and conditional and iterative control structures. Some of the students have gained this knowledge via C or C++, others via Java, and still others via VB 6, or VB .NET. Some department members predicted that with these language-independent prerequisites the course itself was doomed, but the past three years have disproved this prediction, and I believe the reason for this success despite language-independent prerequisites is to be found in the content covered in the course.

4.2 Course Content

To explore the content of the Adv VB .NET course and show the course's emphasis, the content is explored from several different perspectives.

4.2.1 Introduction or Review

Admittedly, the first three weeks of the course are a bit uneven in terms of how students react towards the presented material due to the variety of student programming backgrounds. The Integrated Development Environment (IDE) is covered, as well as the Graphical User Interface (GUI) controls, event-driving programming, and the fundamental programming constructs of VB .NET. Depending on the background of the student, the material is either a fast-paced introduction to the IDE and VB .NET syntax, or it is a review for the student. On purpose, somewhat unusual programming assignments involving timers and random numbers are chosen for this section in order to make the section interesting for students who profess to already know this part of the material.

4.2.2 Object Oriented Application Development

By the third programming assignment, the student is asked to define a class, and create and manipulate objects instantiated from the class. This remains a difficult approach for students, and the programming background of each student begins to matter less, as only very few students are well-versed in object oriented programming. In fact, the hurdle of the course is not in the first three weeks when about half of the students in the course unfamiliar with VB need to learn a new IDE and syntax, but instead the hurdle for the students comes in via Object Oriented (OO) programming and design, an unaccustomed way of thinking for students who have learned procedural programming first or whose idea of OO programming is the use of predefined classes. For the remainder of the semester, all programs contain one or more user-defined classes. An important emphasis of the course is that classes are separated into three different categories which are discussed next.

4.2.3 Three -Tier Design of OO Development

By the middle of the semester, the students are introduced to the three-tier design of OO development. [11]. After introducing the students to the three tiers: presentation, business and data tier, initial emphasis is placed on separating the presentation tier from the business tier; students are requested to separate the code which services the GUI (presentation tier) from the code which takes care of the business logic (business tier). The student learns how to think of the GUI as producing input for the business object and requesting output from the business object. This level of abstraction is rather difficult for students, but it gets at the heart of the

principle of object encapsulation more than any lecture on the topic will. Finally, the last part of the course deals with the data tier.

4.2.4 The Data Tier and ADO .NET

Rather to the surprise of the students, the data tier is introduced with an introduction or review of Microsoft Access, which is unexpected to the students who have previously only encountered simple file handling when programming. The students are requested to assemble an Access database with related tables, and various queries are assigned. The SQL code generated by the queries is examined and Inner Joins are discussed.

Following the review of Access, an introduction to ADO .NET is given, ADO .NET being the set of classes that are used to provide the data access services. Connection, Data Adapter and Dataset classes are examined and the SQL code that was previously generated in Access is now used in the data tier class to generate datasets. Datasets are converted to collections before being passed to the appropriate business object, which in turn will pass the appropriate collection to the presentation class.

4.2.5 Final Project

At the end of the semester, the students are asked to write a final project to present to the entire class. Though this is a distance education class, this still works well. The students send the final project zipped to the instructor as an e-mail attachment. At presentation time the instructor executes each project in the sharing mode of WisLine Web, and the student gets to explain the features of his/her final project to the entire class. We usually also look at interesting parts of the code. The requirements for the final project are seemingly few: it needs to be a multi-form project in 3-tier design, using ADO .NET technology for retrieving and storing data housed in an Access database.

4.2.6 Assessment

The students are graded on the required programming assignments, approximately ten in number over the length of the course, plus the final project. Other requirements include participation in asynchronous discussions and quizzes both offered via Desire2Learn. Three paper-and-pencil tests are given during the semester. The tests are administered at the UW College where the student attends and are administered by the IT staff. The completed tests reach the instructor as pdf e-mail attachments within hours of the given exam.

5 Conclusion

Despite declining enrollment in the CS program, the Adv VB .NET course designed three years ago

successfully survives in the UW Colleges, and has even inspired some students to continue with the core curriculum CS1-CS2-CS3 offerings. Other students have commented that Adv VB .NET was a good preparation for CS1 and CS2.

There are two main reasons for the success of the course. First, distance education made the experiment possible, and the Distributed Classroom model has worked well for the course. Second, the students are interested in a course that offers up-to-date material in a well-known framework such as .NET. The course gives the student a good foundation in object oriented programming, together with an insight into three-tier design. Developing three-tiered object oriented applications that access information in a relational database using SQL is a more realistic programming scenario than the students encounter with simple file writing in other programming courses.

As efforts have been made in the recent years to update the core CS1-CS2-CS3 curriculum [12] which is still primarily offered in C++ or Java, the success of a course such as Adv VB .NET should be born in mind, as it offers insight into potentially successful models of teaching Computer Science.

References:

- [1] Zweben, S., 2003-2004 Taulbee Survey, *Computing Research News*, Vol. 17/No.3, May 2005. <http://www.cra.org/CRN/articles/may05/taulbee.html>
- [2] UW System, Two-Year Campuses <http://www.wisconsin.edu/campuses/twoyear.htm>
- [3] Clear, T., Haataja, A., Meyer, J., Suhonen, J., Varden S., Dimensions of Distance Learning for Computer Education, *Working group reports from ITiCSE on Innovation and technology in computer science education*, (ACM Press, 2001) pp. 101-110
- [4] Instructional Communications Systems, UW Extension. <http://www.uwex.edu/ics/about.htm>
- [5] WisLine Web, UW Extension. <http://www.uwex.edu/ics/wlweb/>
- [6] Desire2Learn, Innovative Learning Technology. <http://www.desire2learn.com/welcome.html>
- [7] Bhagyavati, Kurkovsky S., Whitehead, C., Using Asynchronous Discussion to Enhance Student Participation in CS Courses, *Proceedings of the 36th SIGCSE technical symposium on Computer science education*, (St. Louis, Missouri, 2005) pp. 111 – 115
- [8] Grissom, S., Van Gorp, M., A practical approach to integrating active and collaborative learning into the introductory computer science curriculum, *Journal of Computing Sciences in Colleges* (Volume 16 , Issue 1, October 2000) pp. 95-100
- [9] The Joint Task Force on Computing Curricula, *Computing Curricula 2001: Computer Science*, Final

Report, *Journal on Educational Resources in Computing* (Volume 1, Issue 3es) ACM Press, 2001

- [10] University of Wisconsin Colleges, Catalogue 2005-2007,
<http://www.uwc.edu/academics/catalog.pdf>
- [11] Doke, E., Satzinger, J., Williams S., Douglas D., *Object-Oriented Application Development Using Microsoft Visual Basic .NET*, Thomson, Course Technology, 2003
- [12] Wiggen, T., Computing Curriculum 2001: First-year Options for Computer Science Majors, *Proceedings Midwest Instruction and Computing Symposium*, (Eau Claire, WI, 2005)
<http://mics.cs.uwec.edu/CD%20image/papers/paper8.pdf>