A Case Study Project for Software Engineering Education

Thomas B. Hilburn, Massood Towhidnejad, Sumeera Nangia, and Li Shen

Abstract - This paper advocates the use of the “case study” approach to educating and training software engineers. After an account of the use of case studies in the education of professionals, there is a discussion of problems in educating software professionals and how a case teaching approach can be used to address these problems. The paper describes a project to develop a comprehensive and complete case study, along with supporting educational material. The case study is designed to teach a variety of software topics, modules, and courses: from high school through college and even for continuing professional development.

Index Terms – Software Engineering, Case Study Teaching, Problem Based Learning.

INTRODUCTION

This paper provides a description of an ongoing software engineering education project in the Department of Computer and Software Engineering at Embry-Riddle Aeronautical University. Informally it is called the “Case Study Project”. The project team consists of two professors of software engineering and two graduate students in the Master of Software Engineering program. The initial phase of the project, reported in this paper, lasted for six months, in 2005, and was supported by a grant from the National Science Foundation (“The Network Community for Software Engineering Education” - SWENET - NSF 0080502), which involved the development and distribution of software engineering education modules.

The purpose of the SWENET [1] is to create, collect, and share software engineering course materials within the context of current accreditation guidelines and curricular models. (SWENET material is available at http://www.swenet.org/.) The Case Study project has the same purpose, with the following more specific objectives:

- Produce software development artifacts, which provide a “real-world” basis for teaching software engineering.
- Design and organize the real-world software development artifacts into a set of case modules (mini case studies), which can be used throughout a computing curriculum.
- Deliver software engineering resource materials, which can be adapted to various teaching and learning styles and techniques.
- Engage software engineering educators in the assessment, use, and development of the case study materials.

The shorter term project objective was to create a framework for support of a full case study, which would include the following activities:

- Organize the project and establish process and planning procedures.
- Conduct research into case study teaching.
- Determine a problem domain and select an application in that domain, which would be the subject of the case study.
- Create a foundation scenario for the case study application (e.g., description of a real-world situation that establishes the need for the case application, introduces people who will be part of the case study, and presents conditions that will constrain and guide the application development).

CASE STUDY TEACHING

Case studies were first used in the Harvard Law School in 1871 [2]. Since then, case studies have been a subject of much study and research about their effectiveness in teaching and learning [2]-[6]. They have become a proven and pervasive method of teaching about professional practice in such fields as business, law, and medicine. The term “case study” is used in a variety of ways. In its most naive form, it simply refers to a realistic example used to illustrate a concept or technique. More formerly, and for the purposes of the Case Study Project, a case study involves the application of knowledge and skills, by an individual or group, to the identification and solution of a problem associated with a real-life situation. Such a case study would contain an account (often in a scenario format – a story) of a real-world activity, event, or situation. It might be supported with background material (setting, personalities, sequence of events, and problems and conflicts), artifacts, and data, which is relevant to the situation depicted.

Case studies are meant to encourage participation, debate and understanding. Although they can be used in a didactic, teacher-centered pedagogy, they are most often used in an
active learning, student-centered system where the teacher acts as a facilitator or coach. Case studies are of special value in problem-based learning, which concentrates on the development of problem-solving skills, self-directed learning skills, and team skills [7]. In this project we see a case study as a means to simulate “on-the-job” learning and to expose students to the disorder and uncontrolled nature of real-world problems. There are a number of excellent examples of case studies in science and engineering at the SUNY-Buffalo web site: http://ublib.buffalo.edu/libraries/projects/cases/case.html.

PROBLEMS IN SOFTWARE ENGINEERING EDUCATION

Teaching software engineering in an undergraduate program has two major hurdles:

- Software engineering is a new and emerging discipline. It is not yet mature and some even question whether it is engineering. This partly explains the dearth of material to support teaching software engineering. The ACM/IEEE-CS Guidelines on Software Engineering Education [8] should help advance and support not only the teaching of software engineering, but the development of crucial support materials (e.g., textbooks and web resources).
- Software engineering is a professional field and students need more than courses in fundamentals and theory; they need to learn about and experience professional practice. One of the curriculum guidelines in [8] states that “The curriculum should have a significant real-world basis”. Another guideline states “SE concepts, principles, and issues should be taught as recurring themes throughout the curriculum to help students develop a software engineering mindset”. The question is how to best provide this “real-world” experience and teach “recurring themes” that develop a “software engineering mindset”.

Unfortunately, the use of case studies to teach software engineering has been limited. For example, among the hundreds of case studies in science and engineering at the SUNY Buffalo web site only two concern computer science or software engineering. There are many software textbooks that use case studies to provide examples to illustrate software development concepts and techniques. Although these cases seem to serve the purposes of the books (e.g., discussing planning, requirements, design, or construction issues by using simple examples that are quasi-realistic), they often lack the following:

- Realistic artifacts (often space does not allow providing a complete requirements or design document)
- Completeness (covers only a portion of the life-cycle)
- Ability to decouple from the text and apply in ways not intended by the author
- Techniques for integration into course activities or into the curriculum as a whole
- A scenario format that would motivate students to get engaged in problem identification and solution
- Guidance to the instructor on how to use the case study to teach a course topic or concept.

There are at least two areas in computing that have provided meaningful and productive use of the case study approach. One is the use of case studies to teach computing ethics. There are numerous articles, books and web sites on the teaching of computer ethics using case studies; of particular interest and merit, is the robust example of Epstein’s The Case of the Killer Robot [9].

Another excellent example of the use of the case study concept is the Advanced Placement Program’s Computer Science, Marine Biology Simulation Case Study [10]. It contains artifacts that are realistic and comprehensive, and it uses a scenario-exercise format that is engaging and motivating. The purpose of the case study is to assist in the teaching the AP computer courses and to provide preparation for the AP computer science exam. The nature and style of the AP case study comes closest to the case study project described in this paper.

A CASE STUDY SOLUTION

The Case Study Project focuses on developing case modules, which are related by being part of and derived from a single case, the development of a single software product. Each case module relates to an artifact or activity involved in the development of the product. In addition, each case module is framed as part of a product development narrative, using a scenario format, which involves characters and incidents that might be part of an actual software development project (e.g., formation of a software project team, interaction with upper management, customer and user interviews, writing a use case description, formal inspection of a software artifact, designing a class interface, a design walk-through, system testing, etc.). Although the case study materials concentrate on one domain area and one development approach (process and methodology), it can also serve as an example of how to develop similar case studies, using other domain areas and other development approaches. A case module could be considered a “mini-case study”.

The Case Study Project is intended to cover the complete life-cycle development of a software product (project management, requirement analysis and specification, design, implementation, testing and maintenance). For the initial phase of the case study project it was decided to concentrate on building a foundation for full development: research into case study teaching; identify a case study problem; create a scenario framework; describe a launch of the software development team; develop a software development plan to be used as part of the case study; and develop several related case modules.

Problem Identification

The project team considered and studied a number of possibilities for the case study problem: a computer game, air traffic control software, an academic planner, a math education application, an immune system simulation, a weather reporting system, and the software system for a “Smart House”.

1-4244-0257-3/06/$20.00 © 2006 IEEE

October 28 – 31, 2006, San Diego, CA

36th ASEE/IEEE Frontiers in Education Conference

M1F-2
After consultation with colleagues teaching software engineering at other schools and interactions with undergraduate and graduate software engineering students, the team decided that the best candidate for the case study was a smart house product. For the purposes of the Case Study project, a Smart House is defined as a home management system that allows home dwellers to easily manage their daily lives by providing for a lifestyle that brings together security, environmental and energy management (temperature, humidity and lighting), entertainment, and communications. The Smart House components consist of household devices (e.g., an air conditioning unit, a sound system, a water sprinkler system, etc.), sensors and controllers for the devices, communication links between the components, and a computer system, which will manage the components.

After some additional research on smart home technology, including a discussion with a home builder and personnel working on development of the Duke Smart House at Duke University (http://www.pratt.duke.edu), the team settled on development of a software system that would support a smart house system called DigitalHome.

The DigitalHome System

As described in a case study introductory scenario, the DigitalHome project is part of the vision of the future for a national company HomeOwner, which is the largest national retail chain serving the needs of home owners in building, furnishing, repairing, and improving their homes. The HomeOwner management has decided to develop a prototype DigitalHome (DH) system that has the following features:

- The DH system will allow any web-ready computer, cell phone or PDA to control a home's temperature, humidity, lights, and the state of small appliances.
- The communication center of the DH system will be a personal home owner web page, through which a user can monitor and control home devices and systems.
- The Digital Home will contain a master control device that connects to the home’s broadband Internet connection, and uses wireless communication to send and receive communication between the DH system and the home devices and systems.
- The Digital Home will be equipped with various environment sensors (temperature sensor, light sensor, humidity sensor, power sensor, contact sensor, water sensor, etc.). Using wireless communication, sensor values can read and saved in the home database.
- The DH system includes programmable devices (thermostats, humidstats, and small appliance and lighting controllers), which allows a user to easily monitor and control a home’s environmental characteristics from any location, using a web ready device.
- The DH system includes a DH Planner, which provides a user with the capability to direct the system to set various home parameters (temperature, humidity, lighting, and on/off appliance status) for certain time periods.

The DH Development Team

The DH scenarios describe a set of characters that are involved, directly and indirectly, in the development of the DH system. These include the following:

- HomeOwner management personnel – Robert “Red” Sharpson (founder and CEO), Dick Punch (VP of Marketing) and Judy Fielder (Chief Information Officer).
- A family used in the DH needs assessment – the Wrights (Steve, Mini, Stanley, Vinni, Michelle and Robert)
- The DH development team – a diverse team of professionals with a variety of experience and background. Table 1 contains a description of the team members. The team represents a diverse group of “real people”, who bring different talents and experience to the team. In the case study, this variety is explored and exploited in bringing realistic learning situations to students.

The personnel (management, users and development team) are almost as important to the case study as the artifacts developed. As Demarco and Lister [11] have asserted about people-related problems “they’re more likely to cause trouble on you next assignment than all the design, implementation, and methodology issues you have deal with”. We also believe that integrating realistic personalities into the case study infuses it with authenticity and interest.

<table>
<thead>
<tr>
<th>Role/Name</th>
<th>Bio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Director, DigitalHome Division</td>
<td>Jose Ortiz was born in San Diego, CA in 1955. He received a B.S. in electrical engineering at San Diego State University and subsequently earned an MBA at California State at Transylvania. After working a several aerospace firms in the 1970s and 1980s, Jose went to work as a technical manager at a small IT start up firm, Network Solutions, in 1992. In 2002, Jose was hired by HomeOwner as its Deputy CIO. Although Jose has outstanding management skills and excellent technical knowledge about network administration, IS evolution, and IT architectures, he has little detailed knowledge about software development and recent advancements in software engineering methods and practices.</td>
</tr>
<tr>
<td>Team Leader</td>
<td>Sumeera Nangia came to HomeOwner three years ago, after a decade of work at SoftMedic. At SoftMedic she worked in a variety of roles on the development and maintenance of a number of widely used software applications in the field of health delivery and management. Most recently, at HomeOwner, Sumeera lead the development of an in-house application used by HomeOwner store managers to train new employees. Sumeera is 38 years old and came to the U.S. from India in 1989 to work on a degree in Computer Science at Valley State University. After completing her degree, she went to work for SoftMedic, became a U.S. citizen, and earned an online Master of Software Engineering degree from Lancaster Institute of Technology. She is considered a strong leader and an effective manager, with excellent knowledge and experience in almost all areas of software engineering.</td>
</tr>
</tbody>
</table>

Table 1: The DigitalHome Development Team
### System Analyst

**Michel Jackson**

Michel has worked in the software development business for 30 years. As a new Stanford graduate in mathematics in 1975, he started his career as embedded systems programmer for a major aerospace company. Since then Michel has worked in almost every facet of the software business: as a programmer, a software architect, a system analyst, a quality assurance director, a project manager, and an entrepreneur. He has worked on large, custom built real-time, embedded systems, shrink-wrapped software, and major web-centric applications. Most recently Michel headed a small company that specialized in software analysis and modeling. After selling his firm last year he has been in semi-retirement, with part-time consulting work occupying his non leisure time. His old friend Jose Ortiz has lured Michel out of retirement to work on this project.

### System Architect

**Li Shen**

Li is considered a whiz kid of object-oriented design. At twenty-six years of age he shows great promise to be an innovative leader in software design of embedded consumer products. He was hired by Jose Ortiz right after graduation from Rational University and is viewed as future principal player in the DigitalHome division.

### Software Engineer

**Georgia Magee**

Georgia has worked for the last three years as a programmer and a test engineer at HomeOwner. Prior to that, she had a four year stint with the Volcanic Power Company, as a junior software engineer developing electric power management software. Georgia is married, thirty years old, and recently became a mother with the birth of her first child, George.

**Massood Zewail**

Massood recently graduated from the University of Central California with a joint degree: an undergraduate degree in computer engineering and master’s degree in software engineering. Although Massood was an outstanding student (3.87 GPA) and had two summers of student intern work with MacroSoft Corporation, this is his first full time professional employment.

### Case Study Elements

The long-term plan is to write scenarios, develop DH artifacts, and create processes, data and reports that simulate all of the development team’s work on the DH project. Thus far, three phases (Project Inception, Project Launch, and Planning) have been completed. In these phases the following DH case study documents were developed:

- **Scenarios**
  - Inception Scenario
  - Team Biographical Sketches
  - Launch Scenario
  - Planning Scenario

- **DH Artifacts**
  - Customer Need Statement
  - High-Level Requirement Document
  - DH Context Diagram
  - DH Conceptual Design
  - DH Development Strategy
  - Project Plan (estimates, WBS, schedule)
  - Configuration Management Plan
  - Risk Management Plan
  - Quality plan
  - Process, Data and Reports

- **Launch Scenario**

- **Team Biographical Sketches**

- **High Level Requirements Definition (HLRD)**

- **Customer Need Statement**

- **DigitalHome Conceptual Design**

- **DH Development Strategy**

- **Project Plan**

- **Configuration Management Plan**

- **Risk Management Plan**

- **Quality Plan**

- **Process, Data, and Reports**

- **Launch Scenario**

### Table 2: DIGITALHOME CASE MODULE

| Learning Objectives: Upon completion of this module students will have increased ability to: |
|---------------------------------|---------------------------------|
| • Analyze a customer need statement and the initial set of requirements for a system. |
| • Acquire additional information from a customer about his/her needs. |
| • Work more effectively as part of a team. |

<table>
<thead>
<tr>
<th>Case Study Artifacts:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• DH Customer Need Statement</td>
</tr>
<tr>
<td>• DH High Level Requirements Definition (HLRD)</td>
</tr>
<tr>
<td>• DH Background Scenario</td>
</tr>
<tr>
<td>• DH Team Biographical Sketches</td>
</tr>
<tr>
<td>• DH Launch Scenario</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario:</th>
</tr>
</thead>
<tbody>
<tr>
<td>In early September of 2005, HomeOwner Inc. (the largest national retail chain serving the needs of home owners) established a new DigitalHomeOwner division that was set up to explore the opportunities for equipping and serving “smart houses” (dwellings that integrate smart technology into every aspect of home living). In September and October of 2005, the Marketing Division of HomeOwner conducted a needs assessment for a DigitalHome product that would provide the computer and communication infrastructure for managing and controlling the “smart” devices into a home to best meet the needs and desires of homeowners. The Marketing Division produced two documents: the DH Customer Need Statement and the DH High Level Requirements Definition (HLRD).</td>
</tr>
</tbody>
</table>

Using the documents developed by marketing, DigitalHomeOwner has just launched a pilot project (called the DH project) to examine and assess the technical and developmental issues of the smart house concept. A five person team was assembled for the project and in late October 2005 carried out a “project launch”.

During the launch of the DH Project, Sumeera Nangia, the DH team leader, asked Jorge Ortiz, the DigitalHomeOwner Director, if he could set up a meeting with someone in the HomeOwner Marketing Division to discuss the questions the DH Team had about the HLRD. Jorge contacted the Marketing Division Head and set up a meeting between the DH Team and Karen Mullen, the lead for the DH needs assessment effort.

On the Friday before the meeting with Karen, the DH team meets to plan its interaction with Karen.

### Exercise: |

- As preparation for the case module, ask each student to read the Case Study Artifacts listed above.
- Divide the class into a set of small teams (3-4 people).
- Each team takes on the role of the DH Team and prepares for a meeting with Karen Mullen. The team should carry out the following tasks:
  - Analyze the DH HLRD and discuss any problems or concerns about their understanding of the HLRD.
  - Formulate objectives for the meeting with Karen.
• Make up a set of questions the team would like answered, prior to commencement of project planning and software requirements analysis.
• Assign individual roles for the meeting (e.g., meeting facilitator, taking notes, asking questions, etc.).
• Make up an agenda for the meeting.
• Document there work in the Meeting Preparation Form in the Exercise Booklet.

USING THE DH CASE

For the past four years, one of the authors has used case studies extensively in teaching courses and topics in computing ethics, software project planning, software requirements analysis, team building, design patterns, and software processes analysis. These cases were helpful in teaching “small-scale” software and computing topics, and students were motivated and seemed to enjoy this type of learning activity. However, the case studies addressed issues ranging over a disparate set of problem domains, software engineering practices, and scenario elements; this resulted in students learning about software engineering practices in bits and pieces - there was very little coupling between the case studies and hence no accumulation of scenario experience that allowed progress toward more substantial and complex problems.

As the development of the case study proceeds we plan on using the case artifacts and modules to teach various software engineering courses and topics in the Embry-Riddle’s undergraduate and graduate software engineering programs. We also have requested funding to offer a faculty workshop on case teaching and use of the DH case study material.

CONCLUSION

The case teaching method has been used effectively in many professions (law, medicine, and business) to teach about how to solve problems and make decisions, while dealing with realistic situations, working with practical constraints, and engaging with both human and technical issues. Although the use of the case module in teaching software engineering has been limited, the discipline is prime candidate for such a technique. The Case Study Project described in this paper has the objective of building a framework for using the case module for teaching software engineering. The goal of the DigitalHome software system is to provide a single comprehensive and complete example of the engineering of a software product. In addition, the project provides case modules (mini case studies), which can be used to teach various software engineering topics.

Future work will consist of class testing of the current case study materials and ongoing development of the case study for subsequent phases. It is envisioned that the project will take about three years to complete.

REFERENCES


