Web Engineering Curriculum - A Review of 12 Years of Delivery at Postgraduate Level

Yogesh Deshpande and Athula Ginige

School of Computing and Mathematics
University of Western Sydney
Locked Bag 1797, Penrith South DC, NSW 1797, Australia
{y.deshpande@uws.edu.au, a.ginige@uws.edu.au}

Abstract - Web Engineering was introduced as a specialization in the Master of Information Technology (MIT) course at University of Western Sydney (UWS) in 1999. It has been reviewed internally and by external committees thrice in the last twelve years. These years have witnessed tremendous growth of the Web and Web-based applications. The tools and technologies supporting the Web and Web-applications have mushroomed and methodologies have been introduced in the Web Engineering area. Consequently, the concerns and issues that initially motivated the development of the Web Engineering curriculum have also evolved over time. This paper describes our experience of delivering Web Engineering curriculum, the problems encountered, the solutions to those problems and the lessons learnt.

1 Introduction

The term Web Engineering first appeared in 1997 [1] five years after the Web arrived, described simply as “software engineering for the Web”. It was a reaction mainly to the pell-mell growth of Web sites and Web applications, reminiscent of the software crisis of the 1960s. The definition of Web Engineering as, “the application of systematic, disciplined and quantifiable approaches to development, operation and maintenance of Web-based applications” [2], [3] also was similar to that of software engineering that came out of the 1960s problems. Later, in 2005, the International Society for Web Engineering (ISWE), stressed the centrality of the Web more strongly, describing Web Engineering as “the realization of solutions within the World Wide Web, its applications and its advancement, in particular its approaches, methods, models, principles and tools, which are based on the information and communication technologies of the internet” (http://www.iswe.org): The joint ACM/IEEE Curriculum Review acknowledged Web Engineering as an emerging discipline in December 2008. This paper, therefore, will take for granted that there is a general consensus on what constitutes Web Engineering today and accordingly concentrate on the curriculum issues, without further comment on the definition. This is an important assumption because White [4] has explicitly raised the issue of redefining Web Engineering in the context of the emergence of Web

Science [5]. Zheng [6] also suggests, while providing a historical perspective, that the entire gamut of Web Engineering, "its meaning, definition classification, development methods and techniques", is likely to change in the future. The redefinition of Web Engineering and its effects on curriculum design are left as matters for future discussion. However, it is important to keep this genesis of Web Engineering in mind because it has directly influenced, and will continue to influence in the future, the design of Web Engineering curriculum. It has also had a direct bearing upon the pedagogical approach, as explained later on.

Web Engineering curriculum was first discussed during the first Web Engineering Workshop at the World Wide Web (WWW) Conference, WWW7, in Brisbane in 1998. University of Western Sydney (UWS) introduced the specialisation in 1999. The curriculum issues were subsequently discussed in several forums, including panel discussions at WWW and ICSE (the International Conference on Software Engineering) and elsewhere. Whitehead presented his curriculum at one of the Workshops at a WWW conference, which was subsequently published in the first issue of the Journal of Web Engineering (JWE) in 2002 [7]. The first author of this paper presented a paper on an evolving framework for Web Engineering curriculum in 2004 at ICWE in Munich. Hadjerrouit [8] also has an evolutionary perspective while presenting a pedagogical model of the curriculum. Since then, there have been several informal discussions on Web Engineering curriculum. Mayr [9] focuses on Web project management as a special area although he regards Web Engineering as part of software engineering. Gorgone and Kanabar [10] have a Web-centric curriculum in Information Systems with a strong bias towards non-technical subjects. With the ACM/IEEE imprimatur now, there is a much wider recognition of Web Engineering. The present Workshop on Curriculum, at ICWE2010, is thus a culmination of efforts by many people over more than a decade.

Since its introduction in 1999, UWS has revised the Web Engineering curriculum three times. The revisions reflect the evolution of the Web itself and also the vicissitudes the global economy has experienced during this period. At UWS, Web Engineering and Networking have been identified as the main areas of concern in ICT (Information and Communications Technology) at the master's level.

This paper is a brief review of our experience in creating and delivering the Web Engineering specialisation at master's level over the last dozen years. The paper mainly deals with the curriculum issues, i.e. the content, in the way that Whitehead [7] and Gorgone and Kanabar [10] explore. The pedagogical issues, the overall delivery and the resources required are covered as well, where appropriate, but their detailed treatment is left for another occasion.

The paper is structured as follows. Section 2 deals with the initial work in Web Engineering and the first curriculum delivered at UWS from 1999 for about five years. Section 3 explains the current design with explanatory remarks about the evolution in between the initial curriculum and its latest version. Section 4 gives details of student feedback. Section 5 concludes the paper with a few recommendations for the future.

2 Web Engineering Curriculum at UWS (1999-2003)

Before starting Web Engineering specialisation in 1998-99, UWS had offered various units (the same as courses or subjects in other institutions) at undergraduate level for Web site (and page) construction and interactive applications. They were developed directly out of our and our colleagues' experience in creating and maintaining the faculty Web site and Web-based applications to support teaching, online testing and administration and influenced our pedagogical approach. However, the undergraduate courses essentially concentrated on the technical side. In our own work, we had to deal with effects of organisational boundaries, copyright, reliability of information as well as security and performance problems. These could not be translated successfully into the curriculum at undergraduate level for two reasons. First, with paucity of tools and established methods, the technological details (i.e. Web page and site design and construction, and programming Web applications) took most of the time allotted to the units. Second, the students were not sufficiently mature to understand or deal with organisational and informational problems.

The Web Engineering workshop in 1998 helped to resolve a number of questions about building a curriculum [11]. The following three are still relevant, especially because the Web is not static and Web Engineering continues to evolve with it.

1. What are the knowledge areas of Web Engineering?
2. How can these knowledge areas be combined to build a curriculum?
3. How can future developments be accommodated in building a curriculum?

Without going into the detail of the process followed, the first formulation of Web Engineering curriculum for Master in Information Technology (MIT) course is presented in Figure 1. Both Hadjerrouit [8] and Gorgone and Kanabar [10] have comprehensive lists of the areas for such a curriculum, albeit from different perspectives.

Semester 1			
Web Technology	IT for Virtual Organisations	IT Project Management	Elective
Semester 2			
Web Site management and Security	Web Application Development	IT Project Implementation	Elective

Figure 1: MIT in Web Engineering and Design – Course Structure 1999-2003

The curriculum reflected the main concerns of the time, viz. 1999. There were three Web-related units. Web Technology dealt with the technical side of Web page and site construction. Web Site Management and Security covered information architecture, content management, legal and ethical issues and security and performance of Web sites. Web Application Development was about Web Engineering itself.

From the beginning, our approach to delivering these units was 'constructivist' both in terms of individual and social terms. With constant development of new technologies, standards and tools together as the reality, the teaching staff explicitly acknowledged that they could not be repositories of knowledge that would be transmitted to the students. Students were made aware of this on their own behalf, as part of their reality for the rest of their lives and encouraged to acquire expertise from and independently of the teachers. To this end, they worked solo as well as in groups organised to advance their learning. Thus, Web Technology and Web Application Development were treated more individualistically while Web Site Management and Security and IT Project units required group work.

IT for Virtual Organisation (ITVO) is an interesting case of how the trends in Web development can be captured. The years around 1999 were the dot com boom time and although Amazon.com had not yet become profitable, there was tremendous interest and excitement about new ways of doing business. Virtual organisations were new and innovative. ITVO addressed their opportunities and problems.

IT Project management and IT Project Implementation required students to do projects over one year. Students were able, and encouraged, to experiment with the latest developments, particularly in the Web area, frequently with technologies to which the teaching staff themselves had insufficient exposure. A stand-out case was in 2001, within a very short period after Semantic Web was introduced to the world. Two students undertook a project in Semantic Web, battled through the earlier versions of Microsoft XML parser and other problems, and finally created a working example of Semantic Web for processing student applications involving two hypothetical universities. Their short presentation at the end of the semester was praised for clearly showing the potential of an exciting area that was difficult to grasp through reading and abstract reasoning.

Although there was not much awareness of Web Engineering in the wider community and the student population intent on pursuing higher education in computing and IT, UWS saw a steady demand for the specialisation. Web Engineering students were successful in the job market, some being 'head-hunted' by big corporations. Detailed student feedback from that time, however, was lost during a radical restructuring of the University that lasted a few years from 2001 on.

A few characteristics of the time are worth a brief comment here. The curriculum, described above, reflected the prevailing Web environment but lacked specific methodological underpinnings. New tools and technologies made it possible to create newer categories of applications which could not be simply classified under Web Application Development. Web Engineering community had established itself, generating methodological insights and other, incremental improvements. These factors and, in our case, the restructuring of the University, led to a full review of the MIT course, including Web Engineering specialisation.

The review process brought out the fact that, essentially, the curriculum design looked like an ad-hoc attempt albeit reasonably successful in meeting the requirements we had set out for ourselves, including answers to the three questions mentioned at the start of this section. It became clear that there had to be a logical framework that would last longer than the latest technological development. The discussions led to the maturity model in Figure 2, (reproduced from Ginige, [12]). The model clearly represents the concerns of the time but in an abstract way. There are now many software packages, frameworks, tools and standards that make Web page and site construction relatively easy. The design aspects as well project planning and management cannot be dealt with in the same way. There are also methodological developments in building applications, i.e. Web Engineering.

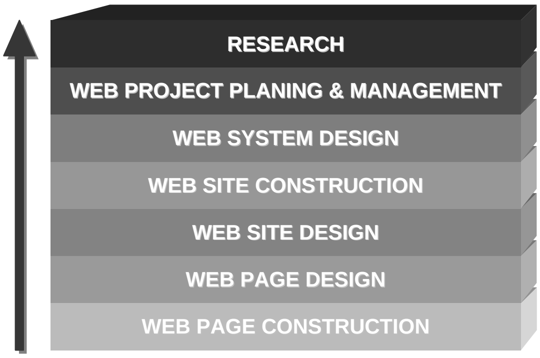

Figure 2: Web System Development Maturity Model

The first review of the curriculum took place in 2002-3. Since then there have been two more curriculum reviews. Rather than narrating a history of those reviews, Section 3 concentrates on the lessons learnt and the revisions to the curriculum made from the time of the first review until now.

3 Current Status of Web Engineering Curriculum at UWS

This section details the important changes we made in the Web Engineering curriculum over the last seven years, and their consequences. The changes can be broadly characterised as methodological, technological and pedagogical improvements. The first two are in terms of content and the pedagogical ones are about the methods of delivery. We also present student feedback over the last four years, in the next section. Together, they lead to the recommendations for future development of the curriculum, which are covered in Section 4. Before outlining the changes, it is useful to introduce a Web Engineering maturity model.

The maturity model, presented in Figure 2, is probably still valid in its essence, although page and site construction are now much easier than before, as mentioned before. The model mainly applies to how an organisation is likely to proceed with its own Web development. It does not cover the processes underpinning those steps. In other words, the model does not attempt to indicate any methodological steps or use of tools and technologies or specific applications, i.e. Web Engineering. To that end, we now have a model depicted in Figure 3.

Research	
Advanced Web Applications (Social Web, Web 2.0)	
Web sites, Mobile Web, Web Apps	**Non-Web site Applications** **(Web Services, component-based applications)**
Web Engineering Methodologies and Project Management	
Internet Technologies	

Figure 3: Web Engineering Maturity Model

The Internet Technologies (layer 1) are the bedrock on which the rest of the model is built. Layer three shows Web sites, mobile Web and Web applications on one side and non-Web site applications on the other, all built on the Internet technologies. The site and application development are not ad-hoc and that is clarified by the intervening layer which represents both Web Engineering methodologies and project management. The bottom three layers may be characterised as the conventional Web or Web 1.0. The fourth layer is Web 2.0. The top layer, Research, is self-explanatory, built on the other four layers.

3.1 The Curriculum in 2010

Figure 4 shows the current structure for the Web Engineering specialisation in the revised Master of Information and Communications Technology (MICT).

Semester 1			
Web Technologies	Web Engineering	Elective	Network Technology
Semester 2			
Content Management and Web Analytics	XML and Web Services	Workflow Management Systems	Elective

Figure 4: Curriculum for Web Engineering Specialisation

3.2 Methodological Change in Curriculum

The major methodological change took place in 2003. Web Application Development was replaced by Web Engineering unit, specifically to remove the ad-hoc nature of application development. The next big change has been the relatively recent adoption of MVC (Model, View and Controller) approach.

3.3 Technological Changes in Curriculum

Web Technology and Human-Web Interaction - At the time of the first revision in 2002-3, we thought that with the spread of the Web, students were generally familiar with the technical side of constructing Web pages and sites. The design aspects on the other hand would always need instruction. Accordingly we discontinued Web Technology and introduced Human-Web Interaction. Over the years, the assumption was proven to be unrealistic. This may be due to changes in the student cohorts, in contrast with some of the earlier ones. We have now restored Web Technology to its previous status. The design aspects are covered at different stages but without a full unit. This is seen as a compromise that can be improved. See the final recommendations, below.

Network Technology – In the original curriculum, Network Technology was available as an elective. Now it is a core unit.

Removal of Web Programming Languages – This unit is not listed in Figure 1. It was an elective then. In the first revision, teaching programming languages was regarded as unsuitable at master's level and it was replaced by Enterprise Web Application Development. In practice, the unit mainly covered Java. It is now discontinued.

XML and Web Services as a unit was added to the curriculum in 2003, in response to the latest development. It has maintained its place.

IT for Virtual Organisations was regarded as not essential to Web Engineering. It has been discontinued.

Workflow Management Systems are seen as important and therefore form a separate unit.

Web Site Management and Security went through a metamorphosis, first as Content Management and Security and now as Content Management and Web Analytics. The syllabus for Security is covered by another unit, in tandem with Network Security and is available as an elective.

Advanced Topics in ICT has been introduced as an elective to take care of emerging topics or topics not otherwise covered. They include Virtualisation, Cloud Computing, Data Mining, Visualisation, and Health Informatics. Green ICT is likely to be added to the list.

In summary, the curriculum reviews have responded to the changing technological and Web landscape.

3.4 Pedagogical Changes

Constructivist approach. There is now much less emphasis on one-sided lectures. Group activities, mini-projects and incremental assignments, e.g. assignment 2 continuing the work done in assignment 1, have become the norm.

Virtual servers and individual domains. In several units, students get their own virtual servers with full administrative rights. These virtual machines are behind their own firewalls, thus protecting them from intruders. Students are also encouraged to get their own domain.

Examinations. Most of the units do not have final examinations. Continuous assessments include quizzes, individual assignments, and mini-projects in groups.

4 Student Feedback

Following restructuring, the University has maintained student feedback centrally from 2005. The feedback is collected on the basis of a questionnaire consisting of 12 statements, with student responses on a Likert scale of 1 to 5 (from 1 for strongly disagree to 5 for strongly agree). Of the 12 statements, six relate to unit content, relevance, learning design, workload, generic skills and overall experience. Their exact wording is as follows.

1. [Unit Content] - The unit covered what the unit outline said it would.
2. [Relevance] - I was able to see the relevance of this unit to my course.
3. [Learning Design] - The learning activities in this unit have helped my learning.
4. [Workload] - The amount of work required in this unit was reasonable.
5. [Generic Skills] - This unit helped me develop my skills in critical thinking, analysing, problem solving and communicating.
6. [Overall Experience] - Overall, I've had a satisfactory learning experience in this unit.

The remaining six questions cover assessments, learning resources and equity. For the present purpose, the six statements enumerated above are regarded as sufficient. Tables 1 to 4 present students' responses over the last four years, for five units, including the discontinued Human-Web Interaction.

Table 1 – Student Responses for the unit Web Engineering

Year	# surveys	received	response	1	2	3	4	5	6
2009	16	15	94%	4.2	4.3	4.3	3.5	4.2	4.1
2008	20	15	75%	4	4.3	4.2	3.6	4.1	3.9
2007	21	21	100%	3.9	4.3	4	3.4	4.2	4.1
2006	38	10	26%	4.1	4.7	4.1	4.4	4.1	4.4

Table 2 – Student Responses for the unit XML and Web Services

Year	# surveys	received	response	1	2	3	4	5	6
2008	24	20	83%	4.4	4.4	4.6	4.3	4.5	4.4
2007	18	15	83%	4	4.5	4.1	4.2	3.8	3.9
2006	29	18	62%	4.1	3.9	3.8	3.8	3.9	3.7
2005	39	16	41%	3.6	4	3	3.5	3.3	3.1

Table 3 – Student Responses for the unit Human-Web Interaction

Year	# surveys	received	response	1	2	3	4	5	6
2007	20	17	85%	4.6	4.3	4.2	3.8	4.3	4.2
2006	38	30	79%	4.3	4	3.9	3.8	4	4.2
2005	37	21	57%	4.4	4.2	3.9	3.4	4	4.2

Table 4 – Student Responses for the unit Web Technology

Year	# surveys	received	response	1	2	3	4	5	6
2009[1]	26	22	85%	4.5	4.4	4.2	4.3	4.1	4.2
2009[1]	32	13	41%	3.4	3.5	3.5	3.3	3.2	3.5

Notes

1. The new form of Web Technology started only in 2009. The two rows correspond to the two semesters in which the subject was taught.
2. Human-Web Interaction was discontinued at the end of 2008. Survey results for 2008 are not available.

4.1 Commentary

Although not reported in the tables, the student numbers have remained fairly steady through these 12 years, with a dip just after the dot com bust. Their composition has also changed from overwhelmingly international to almost equal mix of international and local. Students' performance and responses have also varied over time. Even so, as the Tables show, students rate content, relevance, and overall experience (criteria 1, 2 and 6) in all these units quite favourably. Learning design and workload have more variable responses. In general, each unit is taught by a single staff member with some substitution at times. Learning design is currently decided by each staff member. Over the last several years, there has been a gradual take up of constructivist approach. The tables show improved ratings of the learning design in keeping with this development. As far as generic skills are concerned, the responses have more variations across the individual subjects. There are anecdotal indications that students are more tuned into the technical aspects, regarding other areas as less important. We have not specifically created a unit for legal and other concerns related to the Web and how the general public uses it. Tackling them and getting students to appreciate their importance will be a challenge. It is time for Web Engineering community to consider this aspect seriously.

5 Conclusions and Recommendations

Web Engineering is now an emerging discipline. At UWS, we have had early successes when students were literally struck by the Web's novelty. That novelty seems to have worn off but new technological developments still capture people's imagination. The Web has led to serious concerns about legal, social and other non-technical issues which have to be taken on board. Similarly; Web Engineering must be able to accommodate the development of Web2.0 and the accompanying social Web, to reflect the collective experience of social groups. In addition, we do think that Human-Web Interaction, Web site and application performance, Security must have a strong presence. How all these topics can be managed within the curriculum in a flexible manner is a great challenge before the Web Engineering community. Finally, there is the Web Science. Its effect on Web Engineering is likely to be quite big and significant but must wait for more deliberation and another paper.

References

1. Gellersen, H.-W., Wicke, R. and Gaedke, M. WebComposition: An Object-Oriented Support System for the Web Engineering Life Cycle. Proceedings WWW6 Conference, Santa Clara, CA, USA, 7-11 April 1997, pp 87-96 (1997)
2. Murugesan, S., Deshpande, Y., Hansen, S. and Ginige, A. Web Engineering: A New Discipline for Development of Web-based Systems. Proceedings of the First ICSE Workshop on Web Engineering, International Conference on Software Engineering, Los Angeles, May 1999 (1999)
3. Deshpande, Y., Murugesan, S., Ginige, A. Hansen, S., Schwabe, D., Gaedke, M. and White, B. Web Engineering. Journal of Web Engineering, vol 1, no. 1, pp 3-17 (2002)
4. White, B. The Need for a Re-Definition of Web Engineering Based on Web Science, http://files.me.com/bebowhite/k97sux, (2010)
5. Berners-Lee, T., Hall, W.,Hendler, J., Shadbolt, N., an Weitzner, D.J. Computer Science: Enhanced: Creating a Science of the Web. Science, 11 August 2006, pp 769-771 (2006)
6. Zheng, J. G. Web Engineering Overview: A Historical Perspective. http://jackzheng.net/cubiclh/webengineering/ webengineering.html, retrieved on 19.5.2010 (2006)
7. Whitehead, J. A Proposed Curriculum for a Masters in Web Engineering. JWE, vol 1, no. 1, pp 18-22 (2002)
8. Hadjerrouit, S. Designing a Pedagogical Model for Web Engineering Education: An Evolutionary Perspective. Journal of Information Technology Education, vol 4, 115-140 (2006)
9. Mayr, H. Web Engineering as a Specialization of Software Engineering: Differences in Project Management Education. Systemics, Cybernetics and Informatics, vol 3, no 5, 84-91 (2005)
10. Gorgone, J. and Kanabar, V. (2002) Masters in Information Systems: A Web-Centric Model Curriculum. http://informingscience.org/proceedings/IS2002Proceedings/papers/Gorgo236WebCe.pdf
11. Deshpande, Y. Web Engineering Curriculum: A Case Study of an Evolving Framework. In: Koch, N., Fraternali, P. and Wirsing, M. (eds) Web Engineering, ICWE2004 Proceedings, July 2004 (2004)
12. Ginige, A. (2002) Web Engineering: Managing the Complexity of Web Systems Development, Proc of the Software Engineering and Knowledge Engineering SEKE 2002 Conference, Ischia, Italy, 15-19 July 2002, pp 721-729