

Preface

This book recounts the experiences of academics in Europe and Asia over the past decade in developing and implementing international models for higher education in software engineering that would be thoroughly informed by and oriented to the requirements of the broad software industry. In that the software industry underpins and drives the continuing global revolution in industry, commerce, and society in general, the formation of the highest quality software professionals is of global significance.

The technological and societal changes underway pose fundamental challenges to higher education institutions in every country. Probably the broadest response these institutions have been developing is a deeper and more active involvement with more sections and layers of society than ever before. In particular, the development of a professional industry-oriented approach to higher education constitutes a significant strand of that response.

WHERE TOPIC FITS IN THE WORLD TODAY

A primary objective of industry-oriented software education is the development of highly skilled, highly educated, and highly employable software professionals, capable of developing effective applied software systems of the best quality for the global software industry. They will contribute to the enhancement and further professionalization of the software industry. They will contribute to enhancing the role of the institutions of higher education in enhancing this key industry and industry in general, through industry-institution-society research and development partnerships and a myriad other mechanisms, through enhanced professionalism, through national and international collaborations, through the development of applications throughout the full range of disciplines in the higher education institutions and, indeed, through the appropriate application of the industry-orientation to the disciplines.

TARGET AUDIENCE

It is hoped that the insights and experiences in these chapters will be of assistance to educationalists and academic programme designers across the world involved in industry-oriented software education seeking to produce appropriately skilled graduates for this global industry. They will also help leaders and policy makers in academic institutions and relevant local and national government departments. They will particularly provide useful ideas and guidance to managers of industrial interactions with academic institutions, including recruiters of software graduates, as well as managers of academic interactions with industry.

At the coal-face of programme and curriculum design incorporating purposeful industry-orientation, these chapters will offer examples, lessons, advice, and guidance, including the provision of appropriate quality assurance of the academic, industry, and academic/industrial interface elements of programmes and their integration in practice, to ensure the achievement of international standards of excellence in their graduates entering this globalised industry.

IMPORTANT ELEMENTS IN EACH CHAPTER

This book consists of twelve stand-alone chapters grouped into five broad, sometimes overlapping sections.

Section 1 comprises the introductory chapter 1, which outlines the evolution of the collaboration on industry-oriented undergraduate software programmes between the Dublin Institute of Technology (DIT) in Ireland and the Harbin Institute of Technology (HIT) in China that led to the annual series of China-Europe Symposia on Software Industry-Oriented Education and to the concept of this book. The chapter also provides a review of key elements in industry-oriented higher education and the quality assurance processes required to underpin this at undergraduate and postgraduate programme levels and in postgraduate research. The discussion is set in the context of the fundamental and global significance of software engineering and the software industry to industry as a whole, and to society in general.

Section 2 contains three chapters dealing with the overall theme of operational, integrated, industry-oriented software degree programmes, with emphasis on the measures taken to copper-fastening their industry orientation.

The development, practicalities, and outcomes of the so-called EMERSION project, an international (Irish/United Kingdom/Chinese) collaboration, within the European Union Asia-Link programme (2003-2006), to develop and implement a software industry-oriented undergraduate degree programme within the School of Software in HIT, are described in chapter 2. This collaboration was an extremely active one devoted to adapting an undergraduate programme from DIT, with features from the University of Wolverhampton (UW) for implementation in HIT. This involved an ambitious human resource development of the teaching and administrative staff in HIT, characterised by extensive China-Europe staff exchanges. A fully documented curriculum with new content and industry interactions, specifying teaching, learning, and assessment modes and elaborating a systematic quality assurance system was developed and matched to the cultural and regulatory systems in China. Important spin-offs were a series of research projects in HIT, DIT, and UW evaluating and optimising these processes relating to industry-oriented software education. There were also significant academic developments in the software programmes offered in DIT and UW resulting from the intensive creative work on the project by staff members of these institutions.

Chapter 3 presents a valuable description of the experiences in China and particularly in HIT to devise and establish viable and sustainable co-operation mechanisms between the School of Software and local and national software industries. The mutual benefits to the various stakeholders in software education - higher education institutions, industries, local and national governments, and the students - are explored in detail and shown to argue strongly for active and continuous institution-industry co-operation. The extensive range of related activities already developed by HIT have proven extremely enterprising and, not surprisingly, beneficial and quite inspiring for the students.

The theme of chapter 4 is an adventurous, international (French/Chinese), collaborative Master's degree programme of HIT and the University of Bordeaux 1 (UB1) in France. This draws together

production engineering and software engineering based on the complementary strengths of the two institutions, that of HIT in computer sciences and software engineering and that of UB1 in enterprise modelling and interoperability, and production systems science and engineering. This is a thoroughly international programme in that all students spend two semesters of the four in both China and France. It is also thoroughly industry-oriented in that all students spend two semesters in industrial placements in China, France, or elsewhere in the world.

Section 3 has a selection of six chapters detailing experiences in China, the United Kingdom, and Turkey of developing and delivering software engineering modules with strong industry orientation as elements within general computing degree programmes.

Chapter 5 presents an interesting approach to developing and delivering software engineering curricula with strong industry orientation to produce ethical professional software engineers. A substantial module on software development is described, covering issues such as software quality framework and measurement tools, professionalism and professional conduct, licensing and self-regulation frameworks, social impact, risk analysis, remedies for failures, and other professional aspects of software engineering. Furthermore, the teaching throughout involves a strong element of active student participation, both in groups and individually, including learner-matched pathways and other mechanisms.

The development processes for a progression of industry-oriented software engineering modules for inclusion in general undergraduate computer science programmes in the Wehei campus of HIT is described in chapter 6. These modules cover introductory software engineering, object oriented software design, software architecture and design patterns, and software quality assurance and testing. In the overall process of developing these modules by the teaching team, the sequencing of the material and the relationships between the modules are carefully examined to avoid unnecessary duplication and help ensure smooth connectedness between them. The so-called bottom-up teaching approach chosen also assists in achieving a holistic induction into software. Each class session begins with an introduction to key concepts in a software coding/design assignment that the students then complete in the middle of the session, which ends with a thorough class discussion. A case study is presented relating to the iterative development of the strategic and tactical decisions in the classroom setting, and the resulting software, to sort an integer array in a hands-on process involving active student participation.

The topic of chapter 7 is a module that emphasises the crucial ability of problem-solving for software engineers in industry and advocates the pro-active introduction of a session to develop this ability among the students at an early stage in the undergraduate programme and preparatory to the introduction to programming. The approach advocated is one based on programming robots to solve certain challenging practical problems followed by graphical programming to simulate the same problem-solving tasks implemented earlier with the robots. A number of individual and group case studies are presented, and very high student involvement and satisfaction are reported due to the visual character of the teaching/learning.

A persuasive case for including a detailed, industry-oriented module on structured parallel programming within undergraduate and postgraduate degree programmes in software engineering is set out in chapter 8. Because the computer and telecommunications industries are rapidly developing and exploiting multi-core processing units, the demand of those and other industries for software engineers with parallel programming skills and abilities to fully exploit these devices is also rapidly growing. In response, an introductory module with three strands, parallel architectures, structured parallelism, and parallel programming is described in this chapter.

The theme of chapter 9 also emphasises the importance of active pedagogy and active learning by students, in this case for the development of expertise in the software systems for enterprise resource planning. These systems are used to organise and integrate the full range of business processes, including, very frequently, change management processes. They can encompass a wide range of technical developments as well as organizational and human resource implications. The module developed begins with the initial analysis by the class group of the major factors involved in choosing an appropriate enterprise resource planning software system and proceeds through the physical preparation of the computer laboratory workplace, and the assignment of tasks and schedules to the different student work groups. The next phase consists of producing comprehensive worksheets on the range of functional and technical practices (so-called closed practices) to be managed in the system and then to create the framework for developing new functionalities and practices (so-called open practices). The final phase of the module is the development and implementation of a set of objective evaluations with appropriate weightings. Thus, this chapter introduces a profoundly industry-oriented theme with extensive technical and software content, but also with extensive social functionalities relating to the workforce and the customers of the enterprise.

Chapter 10 has a wide-ranging discussion on the fundamental values and necessity for co-operation between the software industry and the higher education institutions in the provision of software education to the future designers and managers of the complex software systems that will underpin the next generation of industrial and societal processes. The chapter also introduces comprehensive approaches to and preparations for the development of industry-oriented undergraduate and postgraduate software engineering programmes.

Section 4 has one chapter on the key issue of quality assurance in international software education.

Earlier chapters refer in some detail to quality assurance, but Chapter 11 provides a comprehensive description and analysis of the vital need for quality assurance on the international level in higher software education to ensure the production of skilled engineers that can be relied on to produce world-class software for the world market. The presentation is in the context of the increasing internationalization of software education and the increasing requirements of national and international authorities for the assurance of quality of software and therefore of software engineers. The chapter proposes seven prime factors, *motivation, purpose, team, environment, method, fitness, and safety*, which provide a guiding framework for delivering quality and for assessing that quality. It discusses the particular methods used across the globe to achieve these goals, particularly in international inter-cultural programmes, and explains how the seven factors are invaluable in the quality characterization of such programmes.

Section 5 is concerned with e-learning and support tools.

Here, there is one chapter, chapter 12, which describes an ambitious, comprehensive, Web-based system for training software students in IT project management. A special feature of the approach is the development of a personalised competency catalogue, which, together with a parallel reference competency map of the required outcomes (knowledge and skills) of the training course, identifies the gap between these and allows the design of a personalised course for each student. Furthermore, the educational content of the system, which is competency-based, is built into an ontological structure, which allows the development of large numbers of different personalised learning paths. This would have valuable application in undergraduate and postgraduate programmes, and also in a wide range of lifelong learning situations, a vital aspect of the continuing professionalism of software engineering.

HOW THIS BOOK IMPACTS THE FIELD AND CONTRIBUTES TO THE SUBJECT MATTER

This book has a global perspective for the holistic education of the student software engineers and professionals who will construct the next phase of the profoundly global software industry.

Each chapter describes compelling experiences from many countries and derives fundamental lessons on the profound value of software industry-oriented higher education. Most of the chapters define and re-define software industry-oriented education, but generally focus on the software industry as a partner, collaborator, and certainly as an inspiration in constructing the form and content of the education and training being provided.

Significant course modules and complete undergraduate and postgraduate programmes are described in their origins and initial development, in the evolution of their teaching and learning philosophies, in their implementation and quality assurance, and in their evaluation by cohorts of students. The book is full of experiences, practical research investigations, helpful insights and lessons, as well as directions and guidelines in this work. Throughout, the efforts by the developers of these course modules and overall programmes to grow a range of forms of co-operation with the software industry so as to capture the applied real-life nature, and the essence of the industrial practice of software development in the different countries are described in detail.

The central thrust and argument is to inform fellow academics in every country across the world of the value and, indeed, necessity of this approach to the education and formation of the next generation of software professionals who will shape the coming knowledge economy and society. This argument is also directed to the software industry which can and must have a major strategic role in this educational task.

Most fundamentally, the book is dedicated to the students of software and to the development of the most suitable, most fruitful, and most productive educational and training programmes that will enable them to contribute capably, creatively, and with integrity throughout their careers to shaping the global knowledge society.

Matthew Hussey

Dublin Institute of Technology, Ireland

Bing Wu

Dublin Institute of Technology, Ireland

Xiaofei Xu

Harbin Institute of Technology, China