Preface

An introductory course on Software Engineering remains one of the hardest subjects to teach largely because of the wide range of topics the area encompasses. I have believed for some time that we often tend to teach too many concepts and topics in an introductory course resulting in shallow knowledge and little insight on application of these concepts. And Software Engineering is finally about application of concepts to efficiently engineer good software solutions.

Goals

I believe that an introductory course on Software Engineering should focus on imparting to students the knowledge and skills that are needed to successfully execute a commercial project of a few person-months effort while employing proper practices and techniques. It is worth pointing out that a vast majority of the projects executed in the industry today fall in this scope—executed by a small team over a few months. I also believe that by carefully selecting the concepts and topics, we can, in the course of a semester, achieve this. This is the motivation of this book.

The goal of this book is to introduce to the students a limited number of concepts and practices which will achieve the following two objectives:

- Teach the student the skills needed to execute a smallish commercial project.
- Provide the students necessary conceptual background for undertaking advanced studies in software engineering, through courses or on their own.

Organization

I have included in this book only those concepts that I believe are foundational and through which the two objectives mentioned above can be met. Advanced topics have been consciously left out. As executing a software project requires skills in two dimensions—engineering and project management—this book focuses on key tasks in these two dimensions, and discusses concepts and techniques that can be applied to effectively execute these tasks.

The book is organized in a simple manner, with one chapter for each of the key tasks in a project. For engineering, these tasks are requirements analysis and specification, architecture design, module level design, coding and unit testing, and testing. For project management, the key tasks are project planning and project monitoring and control, but both are discussed together in one chapter on project planning as even monitoring has to be planned. In addition, the book contains one chapter that clearly defines the problem domain of Software Engineering, and another chapter that discusses the central concept of software process which integrates the different tasks executed in a project.

Each chapter opens with some introduction and then clearly lists the chapter goals, or what the reader can expect to learn from the chapter. For the task covered in the chapter, the important concepts are first discussed, followed by a discussion of the output of the task, the desired quality properties of the output, and some practical methods and notations for performing the task. The explanations are supported by examples, and the key learnings are summarized in the end for the reader. The chapter ends with some self-assessment exercises.

Target Audience

The book is primarily intented for an introductory course on Software Engineering in any undergraduate or graduate program. It is targeted for students who know programming but have not had a formal exposure to software engineering.

The book can also be used by professionals who are in a similar state—know some programming but want to be introduced to the systematic approach of software engineering.

Teaching Support and Supplemental Resources

Though the book is self-contained, some teaching support and supplemental resources are available through a website. The URL is:

http://www.cse.iitd.ac.in/ConciseIntroToSE

The resources available on the site include:

- The powerpoint presentations for each chapter in ppt format so instructors can change them to suit their style.
- Various templates for different outputs in a project, that can be used for the student project in the course.
- A case study with most of the major outputs of the project.
- Some practice exercises for unit testing and inspections.

Acknowledgments

I would like to express my gratitude to my editor, Wayne Wheeler, who conceived this idea of a concise introductory book and created this opportunity.

I would also like to express my thanks to my wife, Shikha, and my daughters Sumedha and Sunanda for once again bearing with my moods and odd hours.

Pankaj Jalote
New Delhi, May 2008

http://www.springer.com/978-1-84800-301-9

A Concise Introduction to Software Engineering
Jalote, P.
2008, VIII, 288 p. 67 illus., Softcover
ISBN: 978-1-84800-301-9