

Effective Teaching of SE Curriculum with the help of Real World Instructor

Bharti Aggarwal
Northern India Engineering
College
Shastri Park, Delhi-53

Uma Tomer
Northern India Engineering
College
Shastri Park, Delhi-53

Sanjyoti Tarai
Northern India Engineering
College
Shastri Park, Delhi-53

ABSTRACT

Projects are failing more often than they are successful. After analyzing the courses behind, we found that many projects are missing their schedules, exceeding their budgets, delivering software products with poor quality and in many cases even poor functionality. Software development was misunderstood as only coding, whereas it is something before coding and continues even after deployment of software. In this paper we have mention the teaching methodologies to be made to SE education i. e, reducing the gap between research community and what is applied in industry. We have divided our paper mainly in two sections Course I: which will explain the concept of SE to be taught to students by faculty, Course II practical implementation of projects to be done by students under technical expert.

Keywords: Teaching methodologies ,software development, SE curriculum, reducing gap between education and industry.

1. INTRODUCTION

Due to the failure of Projects they are becoming unsuccessful. After certain analysis regarding the courses behind software engineering, it was felt many projects missing their schedules will affect certain areas like their budgets, quality and in many cases functionality .Many software are given to overcome such problems. But main cause lies in the practical real world knowledge of a software engineer graduate. Skill shortage is artificial component behind all this.

Software, when this word covers it was misunderstood that software development is only coding, where as it is something which starts before one actually starts programming and even after the delivery of software, it continues. So actually SE can be defined as the methods, knowledge to create efficient and cost effective solution to design, construction, operation & maintenance of software and associated products [5].

The role of SE education is to transfer technology and to reduce the gap between research community & what is applied in industry. But many real world application using SE concepts does not map with the SE curriculums.

Now as most of the degrees have SE course to convey the importance of software development to the graduates. Even then SE graduating students are lacking the studies which are necessary to create successful software system, as it required by industry.

So in this paper, we designed SE curriculum to be followed by graduates So that they can develop successful careers and to brief them the techniques the companies use to elicit and track requirements and also about their contest of operations[2]

SE curriculum is divided into two courses .Course I mainly introduce initial software engineering concepts and to be taught in lectures by faculty. Whereas Course II focus on special topics of SE and to develop industry level projects. This is taught by as instructor from industry. This paper presents the summarized syllabus of SE to offer SE simulation approach [3][4]

Software engineering is promoting the establishment of theoretical foundations and practical's disciplines can be developed systematically to obtain economical software that is reliable and satisfies the requirements. SE curriculum mostly includes SDLC models, Requirement Engineering, Verification & validation, Software design, Software measurement and metrics, Software testing, coding, maintenances, Re-Engineering, Reliability, Software cost.

When we face a failure in software a procedure is to be followed to meet that. This procedure is called operating procedure which is made up of different instructions[1].

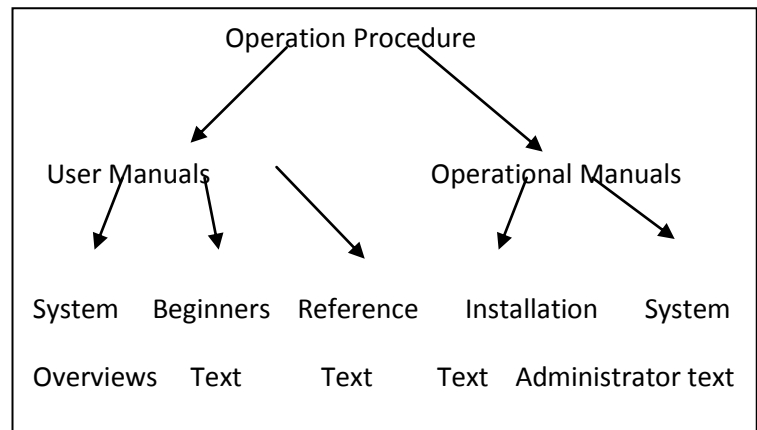


Fig 1: Operating procedures Manuals

By focusing on these different aspects, problems can be sort out. A lecturer gives the theoretical knowledge about all the basic topics of Software Engineering using all the above texts. Whereas the instructor will give brief knowledge about the new topics in Software Engineering and in the end divide the students into different groups for presentation as well as project. Different groups are assigned different tasks or areas of Software Engineering.

2. INTRODUCTORY CONCEPTS OF SOFTWARE ENGINEERING (COURSE-I)

Software Architectures are discussed along with concepts of software design.

Subject faculty /Lectures mainly focus on SE curriculum. The main reason behind to reach this course is to make the students skill perfect so that they can handle multidiscipline areas in the software engineering environment. The curriculum of SE includes in course I is given in table 1. This is to be covered by lecturer.[1]

3. SOFTWARE ENGINEERING SYLLABUS

1. Introduction of SE
2. Software Life Cycle Model
3. Requirement Engineering
4. Verification and Validation
5. Software Design
6. Software Measurement and metrics
7. Software Testing
8. Coding
9. Software Maintenance
10. Re-Engineering
11. Software Cost
12. Software Risk
13. Software Reliability

Topics to be covered by real world instructor are included in course II given below [1]

1. Component Based SE
2. Clean Room SE
3. Software Engineering Support Tools
 - 3.1 Criteria for selecting a tool
 - 3.2 Static/Dynamic Tools
 - 3.3 Specific function tools
4. Software Project Management (PERT/CPM)
 - 4.1 Developing a network plan
 - 4.2 Overview of PERT/CPM
 - 4.3 Basic rules for developing network
 - 4.4 Basic rules for developing project network
5. SE code of Ethics & Social Responsibility
6. Future of SE
7. Case Studies
8. Real Project

This course is mainly intended to give a real world experience to students to develop real projects and let them work according to the software engineering industry[6].

In Course I we mainly focus on brief introduction of SE concepts. This gives the basic knowledge of SE to the students. It lets the students to understand how to develop software systematically and them to maintain it throughout its life. Main motive behind this course is to build various skills in student, like

- General Skill
- Programming Skill
- Communication Skill
- Design Skill

This course mainly gives overview of analysis, evaluation, selection, use of specific systematic approaches of development, operation, maintenance, & retirement of software.

4. PRACTICAL IMPLEMENTATION (COURSE-II)

The intention of SEII is to make the students capable of solving imperfectly specified problem for this; a great amount of engineering design training is required.

The faculty or lecturer for SE I should have following skill sets[1].

- Standards: They could be specific to SE domain is all critical areas of SE
- Principles: Should based on pure computer software principles.
- Models: Development of different models for viewing, simplifying and prototyping.
- Artifacts: Art of documentation and communication for common understanding.
- Tools: Tools to save time and cost.
- Estimation: Estimation of all resources, efforts & inputs for costing & planning.
- Building Standard practice: Use experiences to develop best applications.

In this course special topics of SE are included and in the end, a project is developed by the students in guidance and instructor.

The main process model to be followed by students is given in fig 2.

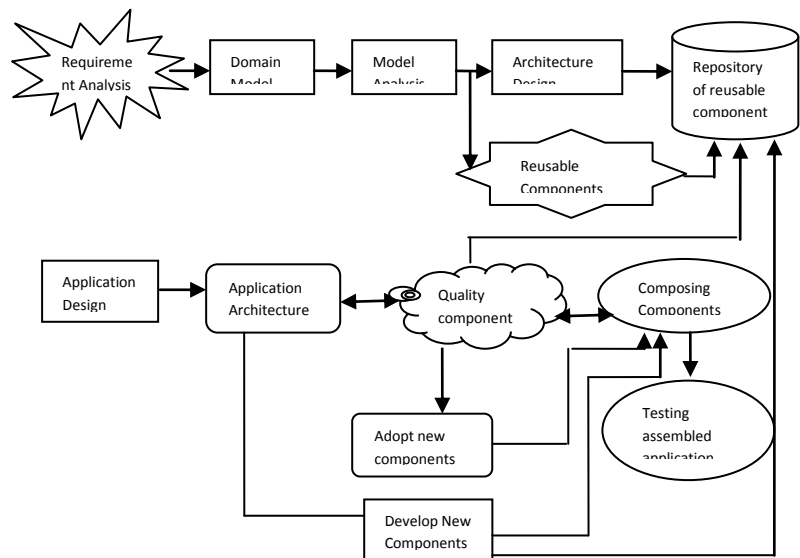


Fig 2: SE Process Model

In special topics the need of CBSE, CSE is explained to students by instructor. CBSE process model is actually based of the feature of reusability. So it is chosen as a software development strategy and is successful .In CSE, all real life topics are developed using SE principles. Box structure is used for text. The uses of different network tools are explained to them. PERT or Gants charts are easy to understand and are cost effective tools. The uses of CASE tools are explained [1].

Some issues and problems from instructor were also put forward so that students can help or guide the instructor to

improve the process. All these are explained practically to the students.

After completion of this section the instructor discuss a problem with some special question along with that for case study, Like:”Provided you are a project manager of a software company and client wants you to develop an application system in a specific time as well as specific budget and needs that on a urgent basis. So what will be your first step in that case, whether you will accept the request or reject it?. If Yes, how will you solve the problem?. All the students are required to give their views after investigating the problem, looking at the problems and conclusion of each step, brought forward different kinds of solutions[1][2].

As a result of this student shows their increased capability for knowledge and tried to solve the realistic problems independently.

After the case study students are asked to develop team presentations. Each team will study a case and then presentation of deliverable updates is given on that case. These presentations and deliverable will define the team and shows how all the members of a team are required to participate in the presentation. The presentation is given in the form of a group discussion among the team members.

And team heads may present the technical work. If the case for study & presentations are not from industry, the instructor will act as a customer, asked them to work.

5. WORKSHOP FOR STUDENTS

In the end of course II, the students are required to develop a project. A single project is divided into different teams of students for project cooperation [4]. Each team has five to six students. Each team will work on different aspect of an application development given in table 1.

Table 1.Application development

Team Work	Deliverable/Method
Team 1: Requirement Analysis & Specification Team	Interviewing, Brainstorming, mind mapping. FAST, JAD, use case scenarios are used for eliciting all requirements.
Team 2: SDLC Team	History of all models studied and reviews to design best suitable model.
Team 3: Project Initiation Team	Development of SRS, study & implement of CBSE, CSE Gantt charts.
Team 4: Design Team	Use case Model, Class diagrams, & modeling. Steps are ; Analyze the problem ↓ Define the design ↓ Evaluate the design ↓ Review the design
Team 5: Software Development Team	Coding & code inspections .Inspection team consists of 4 persons, who play the roles of modulator, reader recorder & author.

Team 6: Testing Team	Test Case, Test Plan & Test Procedure
Team 7: Re-Engineering Team	Redefine software scope & goals. Redefine ROD & SRS Redesign design & architecture. Adapt new technology. Such as new case tools
Team 8: Software	Project Risk Product Risk Business Risk
Team 9: Reliability Team	Reliability, Availability & maintainability
Team 10 : Executing presentation Team	Open Presentation on project to faculty, students, industry instructor

Each team has one team leader who is responsibility for specializing or instructing roles to each student in the team. And the instructor will work as a project manager[7].

Main motive of this project is to encourage the students to simulate a real world development environment utility the concept and methodologies learned in course I and course II. The students are taught to collaborate in subgroup roles and then the subgroups collaborate with other subgroup to accomplish the project successfully. This approach gives a closer view of industry development environments. Team member are integrated and then team are integrated to give a true meaning to integration. The roles are specifically modeled on industry-based roles using latest information technology. Finally, an actual true software product using project is developed.

6. CONCLUSION

After completion of this project, the instructor will give some real project from industry to the student in his guidance. Some of the teams may be from industry also. In conclusion, this paper showed the curriculum division and effective teaching with regular lecturers and with an instructor from industry. It facilitates students understanding of knowledge from industry. Through these effective teaching students are trained to work in real environment, to face real problems and to solve those problems. In this way many shortcoming of software failure can be solved.

7. REFERENCES

- [1] SE by bali bali publisher S.K.Kataria & Sons
- [2] A Student-Enacted Simulation Approach to Software Engineering Education M. Brian Blake, Member, IEEE
- [3] A. Drappa and J. Ludewig, “Simulation in software engineering training,” in Proc. 22nd Int. Conf. Software Engineering, Limerick, Ireland, 2000, pp. 199–208.
- [4] E. Oh and A. van der Hoek, “Challenges in using an economic cost model for software engineering simulation,” in Proc. 3rd Int. Workshop on Economics-Driven Software Engineering Research, Toronto, ON, Canada, May 2001.
- [5] M. B. Blake and T. Cornett, “Teaching an object-oriented software M. B. Blake and T. Cornett, “Teaching an object-oriented software development lifecycle in undergraduate

- software engineering education,” Proc. IEEE Conf. Software Engineering Education and Training (CSEET2002), Feb. 2002.
- [6] M. Moore and C. Potts, “Learning by doing: Goals and experiences of two software engineering project courses,” in Proc. 7th Software Eng. Inst. Conf. Software Engineering Education, San Antonio, TX, Jan. 6, 1994.
- [7] M. Moore et al., “Teaching about the real world: Issues in designing a software engineering practicum,” in Proc. 2nd Software Engineering Research Forum, Melbourne, FL, Nov. 5–6, 1992.
- [8] B. W. Boehm et al., “A stakeholder win-win approach to software engineering education,” in *Annals of Software Engineering*, O. Balci et al., Ed. Amsterdam, The Netherlands: Baltzer Sci., 1998, pp. 295–321.
- [9] The Real World Lab. (2002) Georgia Inst. Technol., Atlanta, GA. [Online]. Available: <http://www.cc.gatech.edu/classes/RWL/Web/>.
- [10] Accreditation Board for Engineering and Technology, Inc. "Engineering Criteria 2000: Criteria for Accrediting Engineering Programs", 1999. <http://www.ele.uri.edu/faculty/daly/criteria.2000.html>
- [11] C.A. Bowers, and D.J. Flinders, *Responsive Teaching*, teachers College Press, New York, 1990.
- [12] R. Creese, "A Project Centered Engineering Program", *Journal of Engineering Education*, ASEE, Washington DC, Vol. 78, No. 2, 1987, pp. 100-104.