CHAPTER 2: IDENTIFICATION OF CORE COMPETENCIES FOR SOFTWARE ENGINEERS

Education programs seek to develop certain generic and discipline specific competencies of students. Educationists, accreditation agencies, professional societies, as well as forums of industry often engage in discourse about the essential and desired competencies as outcomes of education programs. Passow [78] has interpreted the competencies to mean the skills, abilities, knowledge, attitudes, and other characteristics that enable a person to perform skillfully (i.e., to make sound decisions and take effective action), in complex and uncertain situations such as professional work, civic engagement, and personal life. Further, she has viewed expertise as the proficient coordination of multiple competencies that leads to consistently effective performance in a variety of complex, unique, and uncertain situations.

Section 2.1: Study Report on Core Competencies for Engineers with Specific Reference to Software Engineering

We first discuss the various studies related to the core competencies required for general engineering graduates, and come up with the set of general engineering competencies normally accepted among the researchers [78a]. With this set of competencies as a starting point, we did an extensive survey among software engineering practitioners, to find out which subset of engineering competencies are more important for the software engineering graduates.

Bordogna [79] quotes an NSF report (published in 1989) which identified integration, analysis, innovation and synthesis, and contextual understanding as key capabilities for engineering students. He also posits that the essence of engineering is the process of integrating different forms of knowledge to some purpose, and an engineering student must experience the ‘functional core of engineering’- the excitement of facing an open-ended challenge and creating something that has never been. He proposes that a 21st century engineer must have the capacity to:

i. design, in order to meet safety, reliability, environmental, cost, operational, and maintenance objectives,

ii. realize products,
iii. create, operate, and sustain complex systems,
iv. understand the physical constructs and the economic, industrial, social, political,
    and international context within which engineering is practiced,
v. understand and participate in the process of research, and
vi. gain the intellectual skills needed for lifelong learning.

Dodridge [80] classifies the attributes of engineers into two broad categories of (i) knowledge and understanding and (ii) skills. Dodridge (2003) as well as Mason [81] refer to a 1998 survey by the EMTA (Engineering and Marine Training Authority) that identified practical skills, multiskilling, computer literacy, communication skills, management skills, personal skills, and problem solving skills as the most important skill deficiencies among engineers. Hoscette [82] and Erlendsson [83] have identified some workplace defects and leading causes of failures in engineering. As per their observation, the major concerns are passivity, non-responsiveness, uncritical thinking, technical incompetence, inept or poor communication skills, poor relations with the supervisor, inflexibility, poor and lax working habits, and too much independence.

Successful Practices in International Engineering Education (SPINE) is a benchmark study [78a] focusing on the analysis of successful practices in engineering education in ten leading European and U.S. universities including MIT, CMU, and ETH, Zurich. The study attempted to measure the perceived importance and assessment of fifty-one parameters on quality of education, teaching methods, engineering competencies, general professional skills, and aspects of reputation of institute through a quantitative analysis. In the SPINE project, 543 professors of these universities, 1372 engineers and 145 managers of European and US companies were questioned. A summary of their findings is given at Annexure AN11.

We administered a survey among Indian engineers and managers working in Indian and multinational IT companies to obtain their perceptions on the importance of forty-nine parameters of engineering education. For the purpose of our first empirical study [84] conducted in 2004-06, we added two additional general professional competencies: (i) awareness of environmental issues, and (ii) sensitivity towards socio-economic aspects for sustainable technological development.
The abovementioned twenty-three competencies were included in this list. Other parameters on teaching methods, quality of education, and aspects of reputation of institutes were the same as in the SPINE survey. The results of this survey with reference to the teaching methods are discussed in Chapter seven. The other two set of surveyed parameters are not included in this thesis. Respondents were requested to assign numeric ratings to these parameters on a scale of 0 to 10, with 10 being the highest importance in terms of the parameter’s criticality and potential contribution in preparing students for a successful professional career.

Fifty-four experts working in fifteen companies responded. The responding experts had industrial experience ranging from 1.5 years to 35 years with an average experience of 7.5 years, which is inferred to be slightly higher than the industry average, given the average age of employees in the Indian IT industry is only 27-30 years [5]. The Collection of these responses was spread over a period of approximately one year from 2003 to 2004. Table 2.1 provides a brief summary of the survey results about the importance of competencies. More details are provided in Appendix A1.

Table 2.1: Most important engineering and general professional competencies, as rated by Indian engineers and managers working in Indian and multi-national IT companies (2004)

<table>
<thead>
<tr>
<th>No</th>
<th>Competency</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Problem solving</td>
<td>Pivotal</td>
</tr>
<tr>
<td>2</td>
<td>Analysis/Methodological skills</td>
<td>Critical</td>
</tr>
<tr>
<td>3</td>
<td>Basic engineering proficiency</td>
<td>Critical</td>
</tr>
<tr>
<td>4</td>
<td>Development know-how</td>
<td>Critical</td>
</tr>
<tr>
<td>5</td>
<td>Teamwork skills</td>
<td>Critical</td>
</tr>
<tr>
<td>6</td>
<td>English language skills</td>
<td>Critical</td>
</tr>
<tr>
<td>7</td>
<td>Presentation skills</td>
<td>Critical</td>
</tr>
<tr>
<td>8</td>
<td>Practical engineering experience</td>
<td>Critical</td>
</tr>
<tr>
<td>9</td>
<td>Leadership skills</td>
<td>Critical</td>
</tr>
<tr>
<td>10</td>
<td>Communication skills</td>
<td>Critical</td>
</tr>
</tbody>
</table>

Problem solving skill was also identified as the most important competency by the responding engineers in the SPINE project [78a], as well as a University of Arkansas study [85]. Problem
solving is the ability to identify and solve problems, when and where they occur. Domelen [86] quotes Steward (1982), “... all problem solving is based on two types of knowledge: knowledge of problem-solving strategies, and conceptual knowledge.” Gary [87] argues that curriculum should provide opportunities for transforming a problem statement into a model, conjecturing solutions, selecting or developing the appropriate mathematics, examining the analysis, and continuing to transform the conjecture into a solution. Bruner [88] proposed that preparing students for solving real-life problems require a different paradigm of education and learning skills, including self-directed learning, active collaboration, and consideration of multiple perspectives. Problems of this nature do not have “right” answers, and the knowledge to understand and resolve them is changing rapidly, thus requiring an ongoing and evolutionary approach to ‘learning’.

The findings of this study, based on the ratings assigned by Indian engineers and managers working in the Indian and multinational IT companies, as summarized in Table 2.1, are highly compatible with the findings of the SPINE project, which examined the requirements for Europe and the USA in a more general context of the engineering industry. However, importance of development know-how, practical engineering experience, research know-how, and specialized engineering proficiency have been rated at a higher level by the respondents of the current study, as compared to the respondents of the SPINE project. We can explain this difference by examining the nature of the Indian IT industry. This difference may perhaps be partially attributed to the fast obsolescence in the IT industry. Further, the Indian IT industry is mainly a “service industry.” Many companies want to have “industry ready” engineers. Often some companies mention some specific IT skills like the ability to program in specific computer languages, and the use of development tools as recruitment criteria for fresh engineers.

Interestingly, the importance of other language skills has been rated very low as compared to the SPINE rating. As Indian IT companies begin to play a larger role in non-English speaking countries, this is likely to change marginally. Some companies have already started recommending potential recruits to acquire skills in languages like Japanese. There are some noticeable differences with respect to the NASSCOM-KPMG and also Indian Task Force reports
[89] that classified *spoken English, team-working, initiative/enthusiasm,* and *motivation/drive* as desirable skills rather than necessary skills.

Two competencies not examined by the SPINE project, and introduced in this study, were *awareness of environmental issues* and *sensitivity towards socio-economic aspects for sustainable technological development*. The first among these has come out as ‘obligatory’ while the second has been rated as a *desirable* competency.

Hence, we find conclude that the identified core competencies for general engineering graduates were also required by software engineers, but there were major difference in their order of importance.

**Section 2.2: Necessary Competencies as Educational Outcomes for Software Engineers as Recommended by Accreditation Boards, Professional Societies’ and Other Approaches**

Curriculum content is no longer the key as the accreditation agencies in many countries have transformed their accreditation criteria and standards in terms of core competencies. A major shift has taken place from input-based criteria to outcome-based approach. NAE in their vision report for 2020 [15] recommends that engineering schools should vigorously exploit the flexibility inherent in the outcome-based accreditation approach to experiment with novel models for baccalaureate education. Subsequently, we carried out an extensive study of the recommended outcomes by accreditation boards of some countries. We examined the recommended outcomes by Accreditation Board for Engineering and Technology (ABET) (United States) [90-92], United Kingdom Standards for Professional Engineering Competence (UK-SPEC) [93], Institution of Engineers, Singapore (IES) [94], Engineers Australia Accreditation Board [95], and Japan Accreditation Board for Engineering Education (JABEE) [96]. These are summarized in Annexure AN1.

There are great similarities in the competency set identified by the accreditation agencies of the US, UK, Australia, Japan, and Singapore. Nine out of the eleven competencies identified by the ABET, US continue to reappear with some modifications in the competency list prescribed by accreditation agencies of all of these countries. However, some agencies have broadened the
scope of some of these competencies to be more comprehensive. For example, the JABEE has broadened *ability to work in multi-disciplinary teams* into *ability and intellectual foundation for considering issues from a global and multi-lateral viewpoint*, and also has put it at the first position of their list. We considered these competency lists to be ordered on importance as perceived by respective agency. While there are many similarities in the order proposed by these agencies, the JABEE has ordered their list differently. It gives highest importance to *the ability and intellectual foundation for considering issues from a global and multi-lateral viewpoint* and *understanding of the effects and impact of technology on society and nature, and of engineers’ social responsibilities (engineering ethics)*. Table 2.2 gives a summarized and composite view of some of the most commonly distinguished and identified competencies by the ABET, UK-SPEC, EA, JABEE, and IES. *We use these results to further expand and refine our initial set of competencies (Annexure A1) for further investigations.*

**Table 2.2:** Comparative analysis of some common competencies distinguished and identified by some accreditation agencies

<table>
<thead>
<tr>
<th>S.No</th>
<th>Competency</th>
<th>ABET Position</th>
<th>EC2000 Position</th>
<th>UK-SPEC Position</th>
<th>IES Position</th>
<th>EA Position</th>
<th>JABEE Position</th>
<th>Average Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ability to apply knowledge</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1.6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Design skills</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>3.6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Problem solving skills</td>
<td>5</td>
<td>--</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4.25</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Technical competence</td>
<td>11</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>4.8</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Ability to work in multi-disciplinary teams</td>
<td>4</td>
<td>4</td>
<td>9</td>
<td>6</td>
<td>1</td>
<td>4.8</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Sensitivity towards ethical and professional issues</td>
<td>6</td>
<td>5</td>
<td>10</td>
<td>9</td>
<td>2</td>
<td>6.4</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Communication skills</td>
<td>7</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Sensitivity towards global, societal, and environmental issues</td>
<td>8</td>
<td>5</td>
<td>8</td>
<td>7</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Readiness for life-long learning</td>
<td>9</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>7</td>
<td>7.6</td>
<td></td>
</tr>
</tbody>
</table>

**Section 2.2.1: Impact on Curriculum and Future Directions**

The recommendations of the various accreditation agencies in the US, UK, Singapore, Australia, and Japan have already affected educational programs, not only in their respective countries, but also in other countries. Many universities have redefined their program objectives, delivery mechanism, and assessment systems to incorporate graduate attributes in teaching programs [97-97a]. For example, as per National Academy of Engineers (NAE) report, Olin College of Engineering [15] has identified the following characteristics for their graduates:
a. Superb command of engineering fundamentals.
b. Broad perspective on the role of engineering in society.
c. Creativity to envision new solutions to problems.
d. Entrepreneurial skills to bring these visions to reality.

Macro level reforms are being realized through micro level redesigning of every course with a focus on fostering specific competencies [8]. Curriculum now gives more emphasis on design, practice, collaborative learning, humanities, social sciences, and sustainable engineering [98]. Faculty development programs have been organized to help them understand the underlying pedagogical issues [99]. Learning theories and epistemological frameworks are being used to shift the focus of teaching, learning, and assessment processes on competency development [100-100a].

Section 2.2.2: Indian Scenario

One of the nine Indian inventors included in the list of top 100 inventors under 35, Vikram Sheel Kumar, thinks that the biggest challenge an Indian student faces is finding the space to develop an independent mind [101]. Some of the senior industry managers in some industrial sectors feel concerned about the lack of positive attitude, behavioral aspects, ability to cope up with challenges, sincerity, integrity, ethics, self-analysis, discipline, and independent thinking among fresh engineering graduates [102]. It is very ironic that while ‘availability of highly skilled manpower’ has been identified as the most important factor that is driving the increasing momentum of R&D off-shoring/outsourcing industry in India; ‘quality of higher education’ has been identified as one of the main inhibitors [103].

The accreditation criteria defined by the National Board of Accreditation (NBA) of the All India Council of Technical Education (AICTE) [104], has not yet responded to the abovementioned contemporary models that emphasize carefully identified attributes and competencies based on national and global needs. One of the major objectives of NBA is to encourage the institutions to continually strive towards the attainment of excellence. The details of the parameters and their weights as prescribed by the NBA are given in Annexure A1 (Table AN1.1). This clearly shows the NBA is still silent about the core competencies, and continues to assess undergraduate and
postgraduate engineering programs with respect to several inputs rather than focusing and encouraging the institutes to develop a set of carefully identified competencies.

Section 2.3: Some other Contemporary Recommendations About Desired Competencies of Engineering Graduates

According to the Engineering Professors Council (EPC), United Kingdom, the key skills for engineering are communication skills, general IT user abilities, application of numbers, working with others, problem solving, and improving own learning and performance. It also identified the following primary competencies for engineers [105]:

a. Transform existing systems into conceptual models
b. Transform conceptual models into determinable models.
c. Use determinable models to obtain system specifications.
d. Select optimum specifications and create physical models.
e. Apply the results from physical models to create real target systems.
f. Critically review real target systems and personal performance.

The National Academy of Engineers (NAE) suggests that the essence of engineering—the iterative process of designing, predicting performance, building, and testing—should be taught from the earliest stages of the curriculum, including the first year [15]. Further, the NAE [106] has identified the following attributes for engineers of 2020:

a. Strong analytical skills.
b. Practical ingenuity: skill in planning, combining, and adapting.
c. Creativity (invention, innovation, thinking outside the box, art).
d. Communication.
e. Business and management.
f. Leadership.
g. High ethical standards and professionalism.
h. Dynamism, agility, resilience, and flexibility.
i. Lifelong learners.

Rugarcia et al [107] proposed the following categories of necessary skills for engineers:

a. Independent, interdependent and lifetime learning skills,
b. Problem solving, critical and creative thinking skills,
c. Interpersonal and teamwork skills,
d. Communication skills,
e. Self-assessment,
f. Integrative and global thinking skills, and
g. Change-management skills.
Cabrera et al [108] classified the professional competencies for engineers into three main categories of group skills, problem solving skills and professional awareness. The group skills include developing ways to resolve conflict and reach agreement, being aware of the feelings of members in group, listening to ideas of others with open mind, working on collaborative projects as member of a team. The problem solving skills encompass ability to do design, solve an unstructured problem, identify knowledge, resources, and people to solve problem, evaluate arguments and evidence of competing alternatives, apply an abstract concept or idea to a real problem, divide problems into manageable components, clearly describe a problem orally, clearly describe a problem in writing, develop several methods to solve unstructured problems, identify tasks needed to solve an unstructured problem, visualize what the product of a design project would look, weigh the pros and cons of possible solutions to a problem. The third category of professional awareness comprises of an understanding about what engineers do, the language of design, the non-technical side of engineering, and the process of design.

Passow [78] collated some of the earlier research on competencies and expertise in the context of engineering education. She cites Stark et al [110] who surveyed faculty members of nearly 400 universities to find the faculty’s perception of adequate emphasis in different professions, and found that the engineering faculty viewed conceptual competency, as the most important competency closely followed by integrative competency (melding multiple competences to make informed judgments), and communication competency. Professional ethics, technical competence, motivation for continued learning, career marketability, and contextual competence (examining the context from a variety of view points) further expanded this list. This study showed that adaptive competence (propensity of modify, alter, or change elements of professional practice), professional identity, and scholarly concern for improvement were also viewed as reasonably important by responding faculty members.

She has carried out a meta-analysis of twelve empirical studies that had collectively surveyed more than ten thousand engineering graduates about the importance rating of competencies. She has classified the competencies in three groups of top, intermediate, and bottom clusters. The top cluster includes problem solving, communication, and data analysis. The intermediate cluster includes ethics, life-long learning, teamwork, engineering tools, design, and math, science, and
engineering knowledge. The bottom cluster comprises of contemporary issues, experiments, and understanding the impact of one’s work.

Further, Passow’s meta-analysis showed that in addition to the competencies identified by ABET, decision-making, commitment to achieving goals, the ability to integrate theory and practice effectively in work settings, leadership skills, and project management are also extremely important competencies. This study also concluded that respondents from computer science, computer engineering, and software engineering background rated design and engineering tools at a relatively higher level as compared to other engineering disciplines.

We use these recommendations and results to further expand and refine our initial set of competencies (Annexure A1) for further investigations.

Section 2.4: Recommendations of Some International Professional Societies Related to Computing

Recommendations for Computer Science
The Joint Task Force on computing curricula of the IEEE Computer Society and the ACM has published several reports related to computing curricula. These reports make clear recommendations on this issue with reference to specific undergraduate programs in computer science, software engineering, computer engineering, and information technology. The final draft on computing curricula, 2001, suggested the following broad level characteristics of computer science graduates [1]:

a. Systems-level perspective.
b. Appreciation of the interplay between theory and practice.
c. Familiarity with common themes.
d. Significant project experience.
e. Adaptability.

This report also suggested the following general skills for computer science graduates:

a. Communication.
b. Teamwork.
c. Numeracy.
d. Self-management.
e. Professional development.
**Recommendations for Software Engineering**

In 2004, the same task force made specific recommendations about undergraduate degree programs in software engineering [52]. It suggested that graduates of an undergraduate software engineering program must be able to:

- show mastery of the software engineering knowledge and skills, and professional issues necessary to begin practice as a software engineer,
- work as an individual and as part of a team to develop and deliver quality software artifacts,
- reconcile conflicting project objectives, finding acceptable compromises within limitations of cost, time, knowledge, existing systems, and organizations,
- design appropriate solutions in one or more application domains using software engineering approaches that integrate ethical, social, legal, and economic concerns,
- demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for problem identification and analysis, software design, development, implementation, verification, and documentation,
- demonstrate an understanding and appreciation for the importance of negotiation, effective work habits, leadership, and good communication with stakeholders in a typical software development environment, and
- learn new models, techniques, and technologies as they emerge and appreciate the necessity of such continuing professional development.

**Recommendations for Computer Engineering**

In their final report ‘Curriculum guidelines for undergraduate degree programs in computer engineering’ [111], the task force identified the following characteristics for computer engineering graduates:

- System Level Perspective.
- Depth and Breadth (of knowledge).
- Design Experience.
- Use of Tools.
- Professional Practice.
- Communication Skills.

**Recommendations for Information Technology**

In April 2005, the same task force also proposed a draft computing curricula for information technology. This report suggested [112] that pervasive themes for IT program outcome should be *user centeredness and advocacy, information assurance and security, the ability to manage complexity, a deep understanding of information and communication technologies and their associated tools, adaptability, professionalism, and interpersonal skills*. This report also recommends that an IT graduate must acquire the ability to:

- use and apply current technical concepts and practices in the core information technologies,
b. analyze, identify, and define the requirements that must be satisfied to address problems or opportunities faced by organizations or individuals,
c. design effective and usable IT-based solutions and integrate them into the user environment,
d. assist in the creation of an effective project plan,
e. identify and evaluate current and emerging technologies and assess their applicability to address the users’ needs,
f. analyze the impact of technology on individuals, organizations and society, including ethical, legal and policy issues,
g. demonstrate an understanding of best practices and standards and their application,
h. demonstrate independent critical thinking and problem solving skills,
i. collaborate in teams to accomplish a common goal by integrating personal initiative and group cooperation,
j. communicate effectively and efficiently with clients, users and peers both verbally and in writing, using appropriate terminology, and
k. recognize the need for continued learning throughout their career.

**Recommendations for Information Systems**

In 2004, the ACM, Association for Information Systems (AIS), and Association of Information Technology Professionals (AITP) published a joint report on ‘Model curriculum and guidelines for undergraduate degree programs in information systems,’ and characterized this discipline as ‘Technology-enabled Business Development.’ They have divided the representative capabilities and knowledge expected for Information System graduates into the following categories [113]:

a. Analytical and critical thinking: organizational problem solving, ethics and professionalism, and creativity.
b. Business fundamentals.
c. Interpersonal, communication, and team skills.
d. Technology.

do. Technology.

do. Technology.

do. Technology.

**Indian Recommendations**

NASSCOM-KPMG [5] and the Government of India Task Force [114] identify written English, logical reasoning, problem solving and numerical ability, programming skills, listening/empathy, assertiveness and confidence, integrity, values and discipline, sociability, dependability, and reliability as necessary skills for IT professionals. These reports identify spoken English, foreign language, accent understanding, comprehension/creativity, initiative/enthusiasm, team-working, multitasking and time management, and motivation/drive as desirable skills.

It may be noted that the recommendation of NASSCOM as well as that of Government of India Task Force are more influenced by the over emphasized requirements of software service
industry, as shown later in Table 2.4. It sadly ignores the requirements of product development related work in small or large companies.

Hence, we conclude that, not only Indian engineering education accreditation agency, the AICTE (ref: Section 2.2.2), but also the premier trade body and the chamber of commerce of Indian IT industry, NASSCOM, and also the task force created by the central government’s ministry of communication and information technology, have also not yet shown futuristic directions in this regard. The mammoth growth of IT education in India has and continues to take place in an eco-system that is conditioned by serious absence of futuristic vision in the apex institutions.

We use the recommendations discussed in this section to further expand and refine our initial set of competencies (Annexure A1) for further investigations.

**Section 2.5: Some Contemporary Recommendations on Desired Competencies of Software Developers**

The US based Professional Aptitude Council (PAC) conducts a pre-employment aptitude examination for IT professionals. This has also been recently launched in India [115]. This examination consists of questions on nine parameters of problem solving, linear logic, mathematical ability, technical knowledge, applied technical skills, coding skills, creativity, work style, and personality composite. It identifies attention to detail, interpersonal skills, adaptability/flexibility, persistence, sense of urgency, and creativity as IT related personality constructs. Listening, adaptability to new technology, time management, visualize/conceptualize, multi-tasking, business culture, “be the customer” mentality, constructive criticism, organizational skills, stress management, idea initiation, and project management are also highly valued skills in the IT industry [85]. Chang [116] and Erlendsson [117] suggest additional competencies like knowing how to learn rapidly, ability to advocate and influence (persuasion), mentoring, decision making, and ability to manage complexity.

Kelley and Caplan [118] carried out a comparative study of star and average performers at Bell Labs, which showed that taking initiative was ranked as the most important strategy by star
performers, while it was least important for average performers. On the other hand, *ability to give good presentations* was a core strategy for average performers, while it was peripheral for the top engineers.

Turley and Bieman [119] studied the *competencies of software engineers in a Fortune 500 computing company*. They found concern for reliability/quality, focus on user needs, algorithmic and structured thinking, pride in quality/productivity, emphasis on elegant and simple solutions, mastery of skills/techniques, help other, innovative, maintenance of “big picture” view, enjoy challenges, seek help from other, lack of ego, attention to detail, pro-active nature, team orientation, reuse, desire to improve things, perseverance, and strength of conviction are more common competencies of these software engineers.

They further identified that the top 30% software engineers demonstrated significantly higher levels in competencies like help others, pro-active role with management, strength of convictions, mastery of skills/techniques, and maintenance of “big picture” view.

The exceptional software engineers in this study distinguished themselves in terms of their *result orientation* and *sense of mission*, whereas non-exceptional software engineers distinguished themselves in terms of higher perseverance and methodological approach.

They also cited and highlighted the following observations made in earlier behavior oriented software engineering research:

a. The development process was not linear: *designers operated simultaneously at various levels of abstraction and details*.

b. Experienced designers took the users view before proceeding to design. … high-rated systems analysts were more likely to work for a *productive relationship with the users* and specify more requirement than the low-rated analysts. They would reject more hypotheses, try several strategies, apply heuristics, set more goals, and look for analogies to prior problems.
Armour [120] suggested that software developers need *domain specific training*, learning to learn, and structuring mechanism of the representation form.

Connor et al [121] have identified *new and specific technical skill, computer literacy and IT skills, multi-skilling and greater flexibility, the ability to deal with change, an ability to continue learning, re-skilling, and the greater importance of personal and generic skills* as key themes in their assessment of skill trends.

*eXtreme Programming (XP) principles, rules, and practices are based on five core values: communication, simplicity, feedback, courage, and respect* [122]. Shore and Warden have further elaborated upon these values [123] *Communication is aimed at giving the right information to right people when they can use it to its maximum advantage.* Simplicity means to be able to discard unnecessary things. Feedback is to learn the appropriate lessons at every possible opportunity. Courage is required to make the right decisions, even when they are difficult, and to tell the stakeholders when they need to hear it. Respect implies treating oneself and others with dignity, and to acknowledge expertise and mutual desire for success.

Hazzan and Tomakyo [124] highlight the importance of *mental habit of abstraction and the ability to make transitions between levels of abstraction* as an important skill for software developers. Further, relating software engineering to Schön’s work on reflective thinking and professions [125], they also posit that *mental habit of reflection and the ability to move across the ladders of reflections* are closely associated with software engineering processes. Agile methods like eXtreme Programming draw their strength from the possibility of continuous improvement through reflection.

Sodiya et al [126] expanded *Goldberg’s Big Five personality factors* by adding a sixth factor of cognitive ability, and collected the personality traits of nearly 500 software engineers working in different stages of software engineering: requirement engineering, system design, coding, testing/implementation, and delivery/maintenance in Nigeria.
Their findings showed that **agreeableness**: the tendency to be compassionate and not antagonistic towards others, was a universal personality trait among high performing software engineers. This tendency includes being pleasant, tolerant, tactful, helpful, trustworthy, respectful, sympathetic, and modest. The high performing software engineers further showed high levels of **cognitive ability of abstract thinking, analysis, concentration, and visualization**. The other common personality trait among this group was found to be **conscientiousness**: the tendency to be self-disciplined, to be dutiful, achievement and competence oriented, thorough, consultative, and orderly. **Openness to experience**: the tendency to enjoy new intellectual experiences and ideas, imaginative, curious, and broadmindedness was also found to be a common trait of high-performing software engineers, particularly involved in systems testing and integration, management of software process, and deliver/maintenance. **Extraversion**: the tendency to seek stimulation and enjoy the company of others was not found to be a common personality trait of high performing software engineers. **Neurotocism**: the tendency to experience unpleasant emotions relatively easily was found to be universally low among this high performing group.

**Recommendations for Software Architects**

Bass et al [127] have identified that in addition to the knowledge of architectural concepts, software engineering, design, programming, technologies and platforms, the following general competencies are important for software architects:

a. **Communication skills**: Oral and written communication skills, presentation and convincing skills, see and address multiple viewpoints, consulting skills, negotiations skills, understand and express complex topics, listening skills, approachable, and interviewing skills.
b. **Interpersonal skills**: Team player, diverse team environment, creative collaboration, consensus building, balanced participation, diplomatic, mentoring, conflict resolution, respects for people, committed to others success.
c. **Leadership skills**: decision making, initiative, innovative, self-motivated and directed, committed, dedicated, passionate, independent judgment, influential, ambitious, mentoring, coaching, training.
d. **Workload management**: work under pressure, time management, priority assessment, result oriented, estimation, ability to concurrently work well on multiple complex projects and systems.
e. **Skills to excel in corporate environment**: passion for quality, art of strategy, work under supervision and constraints, organizational and work flow skills, process oriented, entrepreneurial, assertive without being aggressive, open to constructive criticism.
f. **Information handling**: detail oriented while maintaining overall vision and focus, see the larger picture, good at working at an abstract level.
g. **Personal qualities**: credible, accountable, responsible, insightful, visionary, creative, perseverant, practical, confident, patient, empathetic, work ethics.
h. Skills for handling unknown and unexpected: tolerant to ambiguity, risk
taking/management, problem solving, reasoning, analytical skills, adaptable, flexible,
open mindedness, resilient, and compromising.

i. Learning: good grasping power, investigative, observation power, adept at using tools.

j. Domain knowledge.

k. Knowledge of industry’s best practices and standards.

l. Knowledge of business practices.

We use the recommendations discussed in this section to further expand and refine our initial set
of competencies (Annexure A1) for further investigations.

Section 2.6: A Perspective from the Professional Codes of Conduct, Ethics,
and/or Practice

Many professions have established professional societies that continuously help and guide their
members to understand their professions not only in terms of technical advancements, but also
evolving understanding of their profession’s context. Professional codes are often designed to
motivate members of an association to behave in certain ways. Codes of ethics are ‘aspirational,’
because they often serve as mission statements for the profession, and thus can provide vision
and objectives. Codes of conduct are oriented more toward the professional, and the
professional's attitude and behavior. Codes of practice relate to operational activities within a
profession. These codes also help them to face and handle professional dilemmas. Primarily,
these codes are designed and used to inspire, guide, educate, and discipline the members. Codes
‘sensitize’ members of a profession to ethical issues and alert them to ethical aspects they
otherwise might overlook. Codes inform the public about the nature and roles of the profession.
Codes also enhance the profession in the eyes of the public. These codes of conduct, practice,
and ethics are not static, and keep on evolving to respond to new challenges and understanding.

All professional societies related to engineering and computing have defined a code of ethics
and/or professional practice. Professional societies like the ACM and IEEE also insist that the
professional education programs must also educate students with these prescribed codes. The
IEEE-ACM joint computing curricula task force on software engineering [52] takes the position,
‘to help insure ethical and professional behavior, software engineering educators have an
obligation to not only make their students familiar with the Code, but to also find ways for
students to engage in discussion and activities that illustrate and illuminate the Code’s eight
principles, including common dilemmas facing professional engineers in typical employment situations.” SWEBOK [68] includes the software ethics under the knowledge area of software quality.

We have examined the codes of conduct, ethics, and/or practice of following societies:

1. American Council of Engineering Companies, 1980
2. National Society of Professional Engineers (NSPE), 1993
3. The Institution of Engineers, Australia
5. American Society of Civil Engineers, 1996
6. American Society of Mechanical Engineers
7. American Institute of Chemical Engineers, 2003
8. IEEE (Institute of Electrical and Electronics Engineers), 1990
9. ACM (Association of Computing Machinery), 1993
10. Information Processing Society of Japan, 1996

The ACM-IEEE Code for Software Engineers Ver 5.2 has eight clauses that address issues related to public, client and employer, product, judgment, management, profession, colleagues, and self. The codes of all the above mentioned societies including ACM-IEEE Code for Software Engineers Ver 5.2, have following common features:

1. The first and the most important recommendation in all these codes is that concerned professional shall fulfill their professional duties by holding paramount the safety, health and welfare of the public. Several clauses of ACM-IEEE Code for Software Engineers Ver 5.2 reflect this concern and objective. These are clause number 1 (1.01 to 1.08), 2 (2.07), and 4 (4.01).

2. The second very important commonly address issue in all these codes is the directive advising their members to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations. Several clauses of the ACM-IEEE Code for Software Engineers Ver 5.2 reflect this concerned and objective. These are clause number 2(2.01), 3(3.04), 4(4.02), 5(5.04), and 7(7.08).
3. The third uniformly occurring instruction to their members is to *act for each employer or client as faithful agents or trustees*. Clause no 2 (2.01 to 2.09) of ACM-IEEE Code for Software Engineers Ver 5.2 expresses this concern in several ways.

4. The fourth identical facet in all these codes is an advice to their members to *issue public statements only in an objective and truthful manner*. Clause no 1 (1.06) and 6 (6.07) of ACM-IEEE Code for Software Engineers Ver 5.2 are expressions of this desired virtue.

5. The fifth regular feature of all these codes is a guidance to *avoid improper solicitation of professional assignments*.

6. The sixth common element of these codes is the suggestion that the members shall continue to *develop relevant skill, knowledge, and expertise throughout their careers and shall actively assist and encourage those under their direction to do likewise*. Clause no 8 (8.01 to 8.06) of ACM-IEEE Code for Software Engineers Ver 5.2 are expressions of this desired trait of software professionals.

7. The seventh common aspect of these codes is about *promoting an ethical approach among colleagues*. Clause no. 5 (5.01 to 5.12) of ACM-IEEE Code for Software Engineers Ver 5.2 are expressions of this desired trait of software professionals.

8. The eighth regular tenet of these codes is guiding the members to *continuously strive for quality, excellence, and adherence to highest professional standards*.

*We use the spirit of these recommendations to further expand and refine our initial set of competencies (Annexure A1) for further investigations.*

**Section 2.7: Classical and Contemporary Recommendations on Desired Competencies of Graduates**

In the above sections, we notice a high emphasis on human and social related competencies that go much beyond the scope of technical competencies. Hence, in order to get a better insight into these aspects from the perspective of university education, in this section, we look at the classical as well as contemporary recommendations about university graduates in general. In the 1850s, a pioneer philosopher of modern higher education, John Henry Newman, wrote a seminal work ‘The Idea of a University Defined and Illustrated’ [128]. As part of this work, he included a
discourse on ‘Knowledge Viewed In Relation To Professional Skill.’ In this discourse, he insisted that

*University training aims at raising the intellectual tone of society,* at cultivating the public mind, at purifying the national taste, at supplying true principles to popular enthusiasm and fixed aims to popular aspiration, at giving enlargement and sobriety to the ideas of the age, at facilitating the exercise of political power, and refining the intercourse of private life. education should give the ability to see things as they are, to go right to the point, to disentangle a skein of thought, to detect what is sophistical, and to discard what is irrelevant ... to fill any post with credit and to master any subject with facility, to accommodate himself to others ... to throw himself into their state of mind, how to bring before them his own, how to influence them, how to come to an understanding with them, how to bear with them, ... to be at home in any society ... [to have] common ground with every class ... [to know] when to speak and when to be silent ... to ask a question pertinently ... [to] be able to converse and gain a lesson seasonably ... [and to enjoy] the repose of a mind that lives in itself, while it lives in the world.

Franklin Bobbitt posited that because of unpredictability of future roles, the curriculum should insist on general education and developing individuals’ intellect rather than just aiming to train them for specific work. He also insisted that education must aim at developing a respect for many of the classic authors of “great books.” These thoughts were also resonated in Robbins Report (1963) [129] that suggested that the purpose of higher education is not simply the “instruction of skills suitable to play a part in the general division of labour” and “the advancement of learning,” but also, “to promote the general powers of the mind ... and transmit ... a common culture and common standards of citizenship.” Martha Nussbaum [130] posited that the purpose of liberal education is to cultivate humanity (world citizenship), and she suggested that to achieve this goal, three capacities need to be cultivated. The first among these is capacity for critical self-examination and critical thinking about one’s own culture and traditions through logical reasoning: consistency of reasoning, correctness of facts, and accuracy of judgment. The second capacity is to see oneself as a human being who is bound to all humans with ties of recognition and concern. The third capacity is for narrative imagination: the ability to empathize with others and to put oneself in another’s place through imagination.

The American Association of College and University [131] has declared the following learning outcomes as essential for all college graduates:

a. Knowledge of human cultures and the physical and natural world by engagement with big questions, both contemporary and enduring
b. **Intellectual and practical skills**: Inquiry and analysis, critical and creative thinking, written and oral communication, quantitative literacy, information literacy, teamwork and problem solving

c. **Personal and social responsibility** through active involvement with diverse communities and real-world challenges: civic knowledge and engagement—local and global, Intercultural knowledge and competence, ethical reasoning and action, foundations and skills for lifelong learning

d. **Integrative learning** through the application of knowledge, skills, and responsibilities to new settings and complex problems

García-Aracil and Van der Velden [132] have proposed their competency classification based on six categories of organizational, methodological, participative, specialized, generic, and socio-emotional competencies. The organizational competencies incorporate working under pressure, accuracy, attention to detail, time management, working independently, and the power of concentration. The methodological competencies comprise of foreign language proficiency, computers skills, understanding social, organizational/technical systems, documenting ideas and information, problem-solving ability, analytical competencies, and learning abilities. The participative competencies encompass planning, coordinating and organizing, economic reasoning, negotiating, assertiveness, decisiveness, persistence leadership, as well as taking responsibilities and decisions. The fourth category of specialized competencies essentially means knowledge of field specific theories and methods. The fifth category of generic competencies include broad general knowledge, cross-disciplinary thinking/knowledge, critical thinking, documenting ideas and information, problem-solving ability, and written as well as oral communications skills. The final category of socio-emotional competencies incorporate reflective thinking, assessing one’s own work, economic reasoning, working in a team, negotiating initiative, assertiveness, decisiveness, persistence, adaptability, leadership, getting personally involved, taking responsibilities, decisions, loyalty, integrity, tolerance, appreciating of different point of view.

Their study showed that the best paid jobs required high levels of participative and methodological competencies, the worst paid jobs emphasized on organizational competencies, and high specialized knowledge contribute to higher wages in some professions like medical science, mathematics (including computing), and engineering. They finally concluded that new emerging work situations require individuals with enhanced levels of participative, methodological, and socio-emotional competencies.
We use the recommendations discussed in this section to further expand and refine our initial set of competencies (Annexure A1) for further investigations.

**Section 2.8: A Comprehensive Distilled View on Desired Competencies**

We have consolidated the abovementioned competencies recommended by engineering accreditation boards, engineering and computing professional agencies (including the code of ethics), and various thinkers of higher education, engineering education, and computing education. These recommendations were have been made with reference to graduates, engineering graduates, and computing graduates. *Appendix A2 gives a comprehensive summary* of these competencies in an alphabetical order of competencies. The importance of so many competencies with reference to software developers education has not been empirically examined in the earlier ranking studies, e.g., SPINE [78a], Bailey and Stefaniak [85], and our own [84]

**Section 2.9: Further Empirical Investigations on Required Core Competencies for Engineering Graduates with Reference to the Indian IT Industry**

Our earlier SPINE based empirical study (Appendix A1) discussed above had its own limitations. It mainly suffered from two deficiencies: (i) The examined competencies were generic in nature that were applicable to all fields of engineering, and these were not grounded in the specific competency literature related to software development. (ii) The software industry was considered a homogeneous entity and did not distinguish between the product based small or large companies and/or large companies mainly involved in offering software services to their clients.

Hence, in 2007, we took another survey. Based on the findings of our first study and various published recommendations about the desired recommendations as proposed by accreditation boards, professional bodies, as well as researchers, we significantly revised and expanded the list of surveyed competencies from twenty-three to thirty-five. Table A3.1 in Appendix A3 maps the competencies of the old (Appendix A1, Table A1.1) and the new list.
Some important competencies, listed in Appendix A2, were still not distinguished in our empirical study, conducted in 2007 (Appendix A3). Some of the important competencies of Appendix A2 that were not examined in 2007 included - curiosity, domain competence, abstraction, algorithmic thinking, knowledge of physical and natural world and intercultural knowledge, reflection, self acceptance and self regulation, and workload management.

Seventy-one experts working in thirteen companies with additions like Accenture, Borland Software, SUN, and TCS responded. The responding experts had industrial experience ranging from 1 year to 22 years, with an average experience of 5.6 years. The data was analyzed in a similar manner to our earlier SPINE-based study. For classification of competencies we added another category at the top to distinguish the topmost recommendation and termed it as ‘Existential.’ Table 2.3 provides the summary of the 2007 results.

**Table 2.3:** Most important competencies as rated by Indian engineers and managers working in Indian and multinational software companies (Revised Study 2007) (More details in Table A3.2, Appendix A3)

<table>
<thead>
<tr>
<th>Category</th>
<th>S.No.</th>
<th>Competency (SNo as per Appendix A2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existential</td>
<td>1</td>
<td>Perseverance, commitment, and hard work (13)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Ability to work in teams (1)</td>
</tr>
<tr>
<td>Pivotal</td>
<td>3</td>
<td>Ability to apply knowledge (2)</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Integrity and authenticity (25)</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Analytical skills (6)</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Accountability and responsibility (25)</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Technical competence (31)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Problem solving skills (22 and 23)</td>
</tr>
<tr>
<td>Critical</td>
<td>9</td>
<td>Listening skills (1)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>Attention to detail (15)</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>Project planning and management (24)</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>Quality consciousness and pursuit of excellence (25)</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>Critical thinking (26)</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>Readiness for lifelong learning (9)</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>Design skills (11)</td>
</tr>
</tbody>
</table>

Table 2.4 enumerates the important competencies of Table A3.2 (Appendix A3) that were rated with higher importance, differently for three different segments of software industry: (i) software services related work at large companies, (ii) product development related work at large or mid-size companies, and (iii) product development related work at small companies. In Section 2.11, we interpret the implications of these findings.
**Table 2.4**: The most important competencies for software development work related to software services and product development

<table>
<thead>
<tr>
<th>Category</th>
<th>Software services related work in large companies (SNo as per Table A3.2, Appendix A3)</th>
<th>Product development work in large/mid-size companies (SNo as per Table A3.2, Appendix A3)</th>
<th>Product development work in small companies (SNo as per Table A3.2, Appendix A3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existential</td>
<td>Ability to work in teams (2)</td>
<td>Ability to work in teams (2)</td>
<td>Perseverance, commitment, and work (1)</td>
</tr>
<tr>
<td></td>
<td>Ability to apply knowledge (3)</td>
<td>Ability to apply knowledge (3)</td>
<td></td>
</tr>
<tr>
<td>Pivotal</td>
<td>Perseverance, commitment, hard work (1)</td>
<td>Accountability and responsibility (6)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ability to apply knowledge (3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Problem solving skills (8)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Research skills (17)</td>
<td></td>
</tr>
<tr>
<td>Critical</td>
<td>Perseverance, commitment, and work (1)</td>
<td>Accountability and responsibility (6)</td>
<td>Attention to detail (10)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Analytical skills (5)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Problem solving skills (8)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Research skills (17)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Readiness for lifelong learning (14)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Technical competence (7)</td>
<td></td>
</tr>
<tr>
<td>Obligatory</td>
<td>Listening skills (9)</td>
<td>Integrity and authenticity (4)</td>
<td>Quality consciousness and pursuit of excellence (12)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Critical thinking (13)</td>
<td>Critical thinking (13)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Design skills (15)</td>
<td>Design skills (15)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Technical competence (7)</td>
<td></td>
</tr>
</tbody>
</table>

**Section 2.10: Classifying the Core Competencies for Software Developers**

**Using Marzano’s Dimensions of Learning for Classifying the Competencies**

*Dimensions of Learning* [138], is a comprehensive model of learning and learning process. It structures the various aspects of learning along the following dimensions: (1) attitudes and perceptions, (2) acquire and integrate knowledge, (3) extend and refine knowledge, (4) use knowledge meaningfully, and (5) productive habits of mind. As per this model, all learning takes place against the backdrop of learners’ attitudes and perceptions and their use of productive habits of mind. Dimension 4 subsumes dimension 3, which in turn subsumes dimension 2. This means that when learners extend and refine knowledge, they continue to acquire knowledge, and when they use knowledge meaningfully, they are still acquiring and extending knowledge.
In 2006, we adapted this model to design a three-dimensional taxonomy of desired competencies. Dimensions 2, 3, and 4 represent different aspects of learning in three hierarchical levels. As there are no orthogonal relations among them, in this discourse, they are merged into one. The new merged dimension can be viewed as having *three internal hierarchical sub-levels*. We suggested that, in essence, there are only *three dimensions of learning*:

a. Dimension 1: Attitudes and Perceptions,

b. Dimension 2: Productive Habits of Mind, and

c. Dimension 3: Acquisition, Integration, Extension, and Meaningful Usage of Knowledge.

Learners’ *attitudes and perceptions* about the purpose of learning, as well as roles of teacher, self, and peers determine their motivation, and very significantly influence depth and performance of their learning.

*Productive habits of mind*: critical thinking, creative thinking, and self-regulation facilitate their learning process.

*Acquisition, Integration, Extension, and Meaningful Usage of Knowledge* is directly manifested in the software developers’ work. It includes competencies like technical competency, problem solving, and communication skills.

The core competencies (for software engineers) studied and identified by us till that time (starting with the set of competencies for general engineers) were mapped in these three dimensions of learning. We posited that attitudes and perceptions affect a professional’s ability to practice. The most important element of education should be to develop required attitudes and perceptions. Under the conditions of the right attitudes and perceptions, professionals use their productive habits of mind to acquire and integrate knowledge. Attitudes, perceptions, and productive habits help them to extend, refine and use knowledge for meaningful tasks. **The first version of our taxonomy was published in 2006** [139] **It is summarized in Table 2.5.**
### Table 2.5: Taxonomy of core competencies for software developers - ver. 1, 2006

<table>
<thead>
<tr>
<th>Dimension 1</th>
<th>Dimension 2</th>
<th>Dimension 3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dimension 1</strong>&lt;br&gt;Attitudes and perceptions&lt;br&gt;(S. No. as per Table A3.2, Appendix A3)</td>
<td><strong>Dimension 2</strong>&lt;br&gt;Productive habits of mind&lt;br&gt;(S. No. as per Table A3.2, Appendix A3)</td>
<td><strong>Dimension 3</strong>&lt;br&gt;Meaningful usage, extension, and acquisition of knowledge&lt;br&gt;(S. No. as per Table A3.2, Appendix A3)</td>
</tr>
<tr>
<td>2. Sense of urgency and stress management (29)</td>
<td>13. Numerical ability (26)</td>
<td>17. Ability to apply knowledge (3)</td>
</tr>
<tr>
<td>3. Adaptability and ability to multi-task (18)</td>
<td>14. Critical thinking (13)</td>
<td>18. Analytical skills (5)</td>
</tr>
<tr>
<td>4. Ability to work in homogeneous, multi-disciplinary, multi-locational, and multicultural teams (2)</td>
<td>15. Creativity and idea initiation (22)</td>
<td>19. Design skills (15)</td>
</tr>
<tr>
<td>5. “Be the customer” mentality (19)</td>
<td></td>
<td>20. Decision making skills (21)</td>
</tr>
<tr>
<td>6. Listening (9)</td>
<td></td>
<td>21. Problem solving skills (8)</td>
</tr>
<tr>
<td>7. Sensitivity towards global, societal, environmental, moral, ethical and professional issues and sustainability (34)</td>
<td></td>
<td>22. Communication skills (16)</td>
</tr>
<tr>
<td>8. Systems-level perspective (including knowledge integration, consideration for multilateral viewpoint, and user-centeredness) (20)</td>
<td></td>
<td>23. Organizational skills (23)</td>
</tr>
<tr>
<td>9. Ability to assist others through mentoring and philanthropic donations (30)</td>
<td></td>
<td>24. Project planning and management (11)</td>
</tr>
<tr>
<td>10. Entrepreneurship (35)</td>
<td></td>
<td>25. Persuasion skills (28)</td>
</tr>
</tbody>
</table>

### Additional Competencies

Four important competencies of Table A3.2 (Appendix A3), later identified by us, were not distinguished in this taxonomy. These were – Integrity and authenticity (No 4 in Table A3.2), Accountability and responsibility (No 6 in Table A3.2), Quality consciousness and pursuit of excellence (No 12 in Table A3.2), and Cost consciousness (No 33 in Table A3.2).

Some other very important competencies, listed in Appendix A2, were also not distinguished in this taxonomy. Some of main competencies of Appendix A2 that were not classified in 2006 included - curiosity, domain competence, abstraction, algorithmic thinking, knowledge of physical and natural world and intercultural knowledge, reflection, self acceptance and self regulation, and workload management.

In Annexure AN3, we briefly discuss the details of some others models about classification of competencies. These include Bloom’s taxonomy of educational objectives [133], Anderson and Krathwohl modification of Bloom’s taxonomy [134], Costa’s model of intellectual functioning [135], Kennedy’s four perspectives on professional expertise [136], The classification of college

60
graduate’s competencies as proposed by Stark et al [137], Marzano’s revised taxonomy [140], earlier classifications cited by García-Aracil and Van der Velden [132], and Kelly Coate [141] schema for curriculum design

In Section 3.11, we will discuss a revised version of our taxonomy of competencies.

**Section 2.11: Chapter Conclusion**

The overall findings of the revised study, as summarized in Table 2.3 and Appendix A3, gave new insights into the importance of desired competencies in software industry. The respondents gave highest importance rating to many newly added competencies that related to attitude and values rather than skill or knowledge. These include perseverance, commitment, and hard work, integrity and authenticity, accountability and responsibility, quality consciousness and pursuit of excellence, “be the customer” mentality, and systems-level perspective. Similarly newly added generic cognitive skills of attention to detail, critical thinking, decision making skills, and creativity and idea initiation were also rated very high by our respondents. Very interestingly, contrary to the popular interpretation of communication ability, listening skill was rated much higher than the communication, presentation, or persuasion skills.

Further, the findings of the revised study, as summarized in Table 2.4, are especially useful for curriculum designers and computing faculty.

For the *Software Services Industry*, the ranked list of top competencies recommended were: (i) ability to work in team, (ii) abilities related to perseverance, commitment and hard work, and (iii) listening skills. Interestingly, all these competencies require development of attitude and perspectives, usually not the focused goal of the commonly prevailing academic process.

For *large and mid-size IT Product Development Industry*, the required pivotal/critical competencies were: (i) ability to work in teams, (ii) ability to apply knowledge, (iii) abilities related to perseverance, commitment and hard work, (iv) accountability and responsibility, (v) analytical skills, (vi) problem solving skills,
and (vii) research skills. Here again, all these competencies also relate to attitude, perspectives, and thinking habits, that are usually not focused upon the commonly prevailing academic process.

For small IT Product Development Industry, the required pivotal/critical competencies comprise of all that are required for a large product company (with some minor change in their ranks), along with a few additional critical competencies: (i) attention to detail, (ii) readiness for lifelong learning, (iii) quality consciousness and pursuit of excellence.

It clearly indicates the nature of the gap which needs to be filled. These finding create a strong case for overhauling the software development education system in every aspect. The educational programs have to be conceptualized very differently from the training programs. Education has two goals of nurturing as well as training. Webster defines ‘educate’ as “to develop mentally, morally, or aesthetically especially by instruction,” “to provide with information,” and also “to condition to feel, believe, or act in a desired way.” Hence, the education, especially higher education, is expected to help in growth of human beings to advanced levels. Training is concerned with development of ‘skills.’ Education on the other hand, has a wider goal of cultivating ‘valuable competencies’ to develop wise and competent professionals and citizens.

It is not sufficient to only aim to train technically skillful software engineers. The education system has to aim to develop competent software development professionals. Consequently, while development of skill and technical knowledge is certainly important, the development of attitude, perspective, and thinking ability is even more important. It is also imperative to understand that these learning outcomes can be achieved mostly through changes in academic process, and also inclusion of a few additional courses.

Further, the findings of the revised study, as summarized in Table 2.4, give even more interesting inputs, especially for educators in India, where the software service industry is currently dominating the software industry. Table 2.4 also shows that the competency needs for the usual
work in very large companies, who are currently the largest recruiters from engineering campus, are very limited. Because of very high visibility and recruitment potential, these companies are currently in a position of influencing the management of educational institutes. The finding of Table 2.4 show that if Indian software educators try to orient the goals of their educational programs for this sector, their students will not be suitable for the other two sectors that are growing silently. Based on our finding, we take a position that in order to inculcate excellence, the educational community should create more partnerships and communication channels with the companies that are involved in product development in large, mid-size or even small sector.

We also need to educate our students that the software industry is not monolithic, and the most dominant voice is not the most futuristic voice. With increasing pressures on profit margins in the post-recession period, and fast growing software service industry in many other countries, we cannot hope to run our software industry solely as a service industry with the current nature of less challenging low cost work. The most natural allies for educational institutes, that will help us bring excellence by being more demanding users of our product, i.e., students, are small sector product development companies. There is an increasing trend of start-up companies. The educational institutes should create partnership and even facilitate their growth. The students also need to be motivated to aspire to work for such companies, and prepare themselves accordingly. How to forge such partnerships and communication channels is beyond the scope of this dissertation.

Further, in the light of several other identified competencies and deeper reflections about learning, we have recently revised our thinking about this classification scheme as well as the core competencies. We believe that these dimensions have interdependence and are not orthogonal. In our new taxonomy, we do not consider our three categories as independent dimensions. As there is an inter-dependence of these categories, we model these competencies as a three-tier taxonomy of twelve core competencies. The details of this revised and comprehensively distilled taxonomy are discussed in the third chapter after discussing the distinguishing features of software development.