Asia-Pacific
Economic Cooperation

2008/SMEWG/SYM/016
Agenda Item: 8.1

Issues on Quality-of-Services and the Role of Training

Purpose: Information
Submitted by: Vietnam

APEC Symposium on Improving Market Access for ICT Outsource SMEs
Hanoi, Vietnam
27–29 October 2008

Issues on QoS in Outsourced Projects

The Rôle of Training

Dang Van Hung

College of Technology
Vietnam National University, Hanoi

Why Outsource?

- to maximize your revenue
- to minimize your expenses
- to get access to specialized skills and services
- to concentrate more on your core business
- to save on money, time and infrastructure
- to improve **customer satisfaction**
- etc.

ICT Outsourcing

- Information and Communication Technology (ICT) is one of the most popular areas of outsourcing
- ICT involves a range of issues: software, equipment, premises, people, third party agreements, and so on
- The outsourcing transition plan should include the Service Level Agreement (SLA) and the outsourcing contract.

Outsourcing Boom

- ICT job outsourcing to India, China and many other countries
- Quality of ICT products is increasing
- Economy of outsourcing countries has increased tremendously after outsourcing (UK, Norway, Australia, US)

QoS: Outsource Main Criticism

- QoS: Two-thirds of the companies that responded to a survey by Information Week reported **either no change or a worsening in customer satisfaction** as a result of business-process outsourcing.
- Training people alone would cost more money: Not only to teach people the actual process of what they have to do, but also to teach them about the company, how business is done in the outsourcing countries, and how to interact with customers
- Saving cost from the contractor sides may reduce the QoS and customers' satisfaction

QoS Improvement

- QoS Improvement is the critical point for the success of outsourced projects
- QoS Improvement includes
  - Internationalization, localization, companies' culture
  - Standardization
  - Security
  - Advanced technology

Component-based Approach for Business Systems!

Component-based Approach

- A system as a set of components
- A component forms a unit of composition with **contractually specified interfaces** and explicit dependencies
- Components interact via their interface
- The approach responds to the increasing demand on QoS and system evolution

Engineering of CB Systems

- begins with the establishment of requirements for the system, and then an architectural design.
- examines requirements to determine what subset is directly amenable to **composition** rather than construction (rather than moving immediately into more detailed design tasks). Doing by asking for each requirement:
  - are commercial off-the-shelf (COTS)?
  - are internally developed reusable components available to implement the requirement?
- to develop those new components meeting the requirements that cannot be implemented with COTS

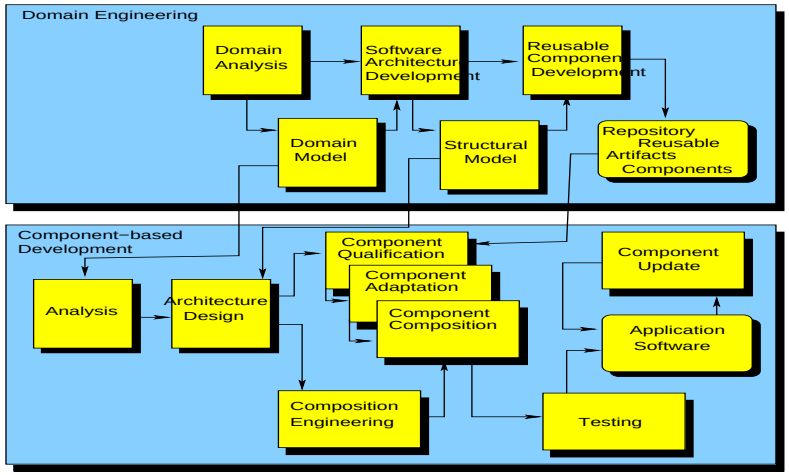
Outsourced IT Project as CBSE

- **CBSE:**
  - different sets of software engineering activities applied to construct new components and to adapt available ones
  - formal techniques to specify component interface (contracts) and to verify if the component is implementing interfaces
- **Outsourced IT Project**:
  - activities to look for suitable partners to construct new components and to adapt available ones to the project
  - techniques to make SLA (Service Level Agreements)

Software Components

- **Qualified components**–assessed by software engineers to ensure that not only functionality, but performance, reliability, usability, and other quality factors conform to the requirement of the system or product to be built
- Adapted components–adapted to modify unwanted or undesirable characteristics
- **Assembled components**–integrated into an architectural style and interconnected with an appropriate infrastructure that allows the components to be coordinated and managed
- **Updated components**–replacing existing software as new versions of components become available

The CBSE Process



Principles Migrated from CBSE

- Component – a nontrivial, nearly independent, and replaceable part of a system that fulfills a clear function in the context of a **well-defined architecture**
- Business component – the software implementation of an autonomous business concept or business process should also have **well defined interfaces**
- Domain engineering to identify a well-defined architecture and components to be implemented with outsourced partners

Principles Migrated from CBSE

- Clear specification of component interfaces (as contracts)
- A contract should include:
  - functionality of services as mapping from required interface to provide interface,
  - **all factors form the QoS** of the business concern
- Component qualification (the required functionality met, fitting into the architecture, exhibiting the quality characteristics)
- **Easy-to-adapt to the requirement changes** should be an important factor of the quality of Architecture and Components

ICT SME Capacity Building

- Capacity to ensure software quality for products in their business area
- **Software Quality Assurance (SQA)** involves the entire software development PROCESS - monitoring and improving the process, making sure that any agreed-upon standards and procedures are followed, and ensuring that problems are found and dealt with. It is oriented to 'prevention'.
- Capacity to train their staff with advanced technologies to follow SQA

Main Quality Attributes?

Availability
Efficiency
Flexibility
Installability
Interoperability
Maintainability
Portability
Reliability
Reusability
Testability
Usability
Performance
Security

Quality Standards

Quality Standards and Auditing Organizations

- SO (International Organization for Standardization).
- CMMI (Capability Maturity Model Integration).
- IEEE (Institute of Electrical and Electronics Engineers).

ISO 9001:2000 (What)

ISO 9001 is a series of documents that define requirements for the Quality Management System Standard. ISO 9001 is one of the documents in this set; it contains the actual requirements an organization must be in compliance with to become ISO 9001 Registered.

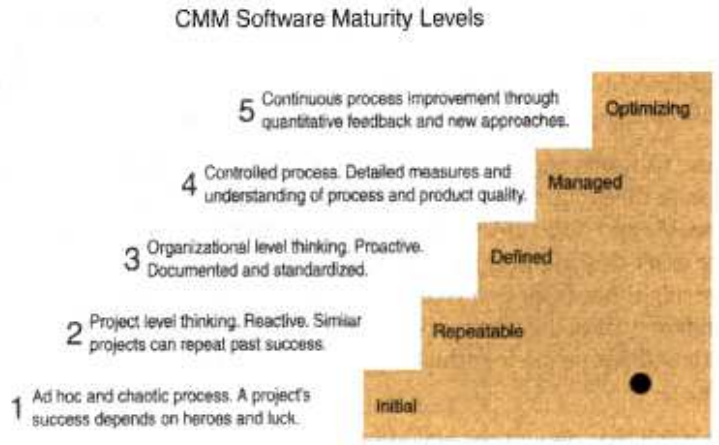
ISO 9001:2000 (Why)

Why Do Companies Want ISO 9001:2000? Many organizations decide to Implement ISO 9001 and obtain registration because it assures customers that the company has a good Quality Management System (QMS) in place. An organization with an effective QMS will typically meet customer expectations better than an organization that does not have an effective QMS. Many organizations require their suppliers to have ISO 9001 Registration.

What Is CMMI?

Capability Maturity Model Integration (CMMI) is a process improvement approach that provides organizations with the essential elements of effective processes. It can be used to guide process improvement across a project, a division, or an entire organization. CMMI provide guidance for quality processes, and provide a point of reference for appraising current processes.

The 5 Levels of CMMI?



The 5 Levels of CMMI

Level One Initial Company has no standard process for software development. Nor does it have a project-tracking system that enables developers to predict costs or finish dates with any accuracy.

Level Two - Managed Company has installed basic software management processes and controls. But there is no consistency or coordination among different groups.

Level Three Defined Company has pulled together a standard set of processes and controls for the entire organization so that developers can move between projects more easily and customers can begin to get consistency from different groups.

The 5 Levels of CMMI

Level Four Quantitatively Managed In addition to implementing standard processes, company has installed systems to measure the quality of those processes across all projects. Âă

Level Five - Optimized Company has accomplished all of the above and can now begin to see patterns in performance over time, so it can tweak its processes in order to improve productivity and reduce defects in software development across the entire organization.

What Is IEEE?

The Institute of Electrical and Electronics Engineers (IEEE) is a non-profit organization that develops, defines, and reviews electronics and computer science standards.

Rôle of Training

A ICT SME should look at itself and see where the organization currently stands, and where it wishes to stand, with repsect to the IT service.

- Time and cost for training staff with advanced Software Technology
- Technologies like Component-based Software Development, Formal Techniques, standard software process should be taught in Universities and retaught in SME
- Fundamental theories to make one selves adapted easily to a new technology

Conclusion

- QoS is a critical issue in outsourced IT projects
- Both the Business System Architecture and Components are contributing to QoS
- CBSE approach to improve QoS
- SME should be aware of SQA
- Training staff with new Technologies and Fundamental Theories as well as skills to follow SQA

Asia-Pacific
Economic Cooperation

2008/SMEWG/SYM/017
Agenda Item: 8.2

Collaboration in Research and Software SME for IT Training – Experiences of IT Faculty, Hanoi University of Technology

Purpose: Information
Submitted by: Vietnam

APEC Symposium on Improving Market Access for ICT Outsource SMEs
Hanoi, Vietnam
27–29 October 2008

APEC Symposium on Improving Market Access for ICT Outsource SMEs

Collaboration in Research and Training for Enhancing Training Quality

Sharing Experiences of IT Faculty, Hanoi University of Technology

Assoc. Prof. Huynh Quyet Thang

Dean of Faculty of Information Technology

Email: thanghq@it-hut.edu.vn

Phone: 844-38692463

Office Address: Hanoi University of Technology, C1 Building, room 327

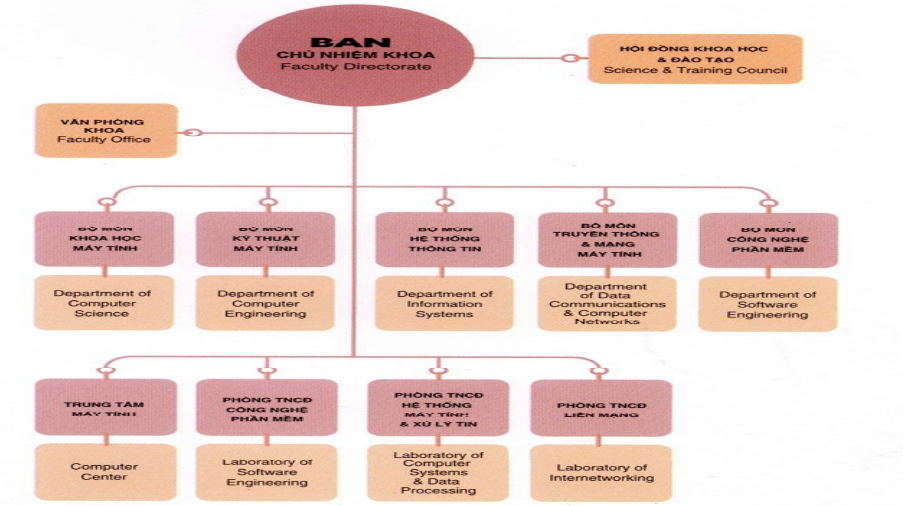
Content

- Introduction of Faculty of Information Technology (FIT)
- Training activities at FIT
- Improving the quality of training
- Research Activities at FIT
- Improving the level of collaboration in research with industry
- Proposals and Conclusion

Introduction

- **Faculty of Information Technology (FIT) of Hanoi University of Technology (HUT)** is one of the leading IT faculties in Vietnam.
- **FIT was established in 1995** by merging three HUT's departments
- **FIT has 5 departments, 1 computer center, the laboratories.**

FIT's Organization Schema



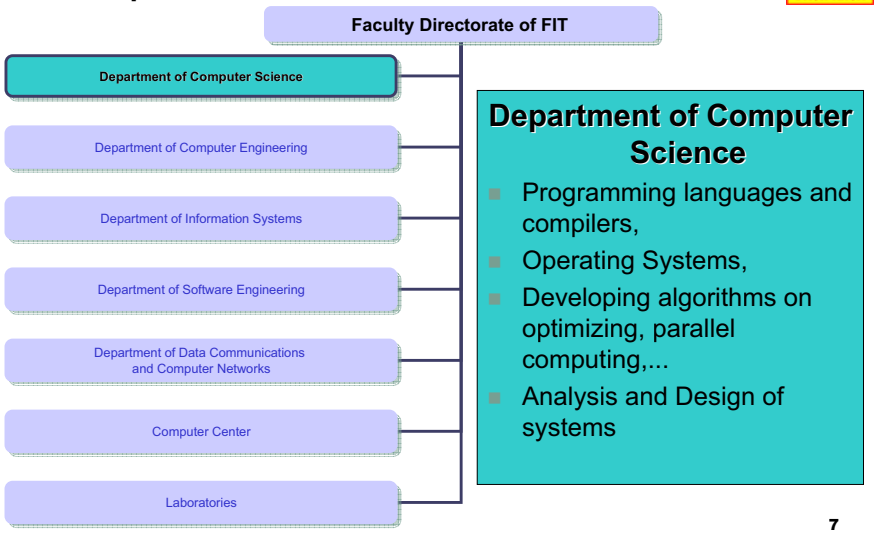
Organization

- **FIT has 5 departments, 1 computer center, the laboratories.**
- **Departments are responsible for:**
  - Teaching, formation of undergraduate degree, of master degree, PhD degree.
  - Scientific research projects
- **Computer Center participates in:**
  - Training, research,
  - Providing services to practice with computers for whole university students.
- **Laboratories support training and research activities of the departments.**

Staff of FIT

- Total staff of FIT- 9/2008: 115 members
- Teaching staffs: 95
  - Professors + Associate Prof.: 7
  - PhD: 27
  - MSc (including PhD students): 46 (22 PhD and Master students in foreign country)
- Support service and technician staff: 17
- Admin. Staffs: 3

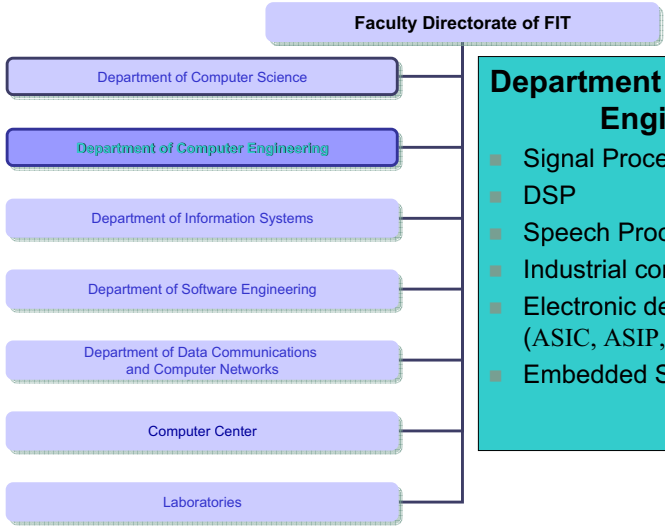
Departments'research areas



Department of Computer Science

- Programming languages and compilers,
- Operating Systems,
- Developing algorithms on optimizing, parallel computing,...
- Analysis and Design of systems

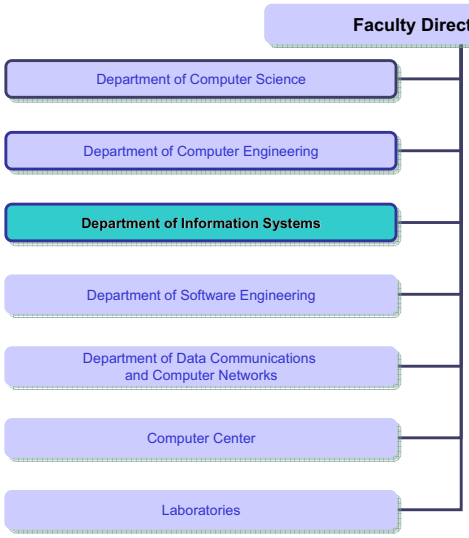
Departments' research areas



Department of Computer Engineering

- Signal Processing,
- DSP
- Speech Processing
- Industrial computer systems,
- Electronic design: chips (ASIC, ASIP, FPGA), circuits..
- Embedded Systems,..

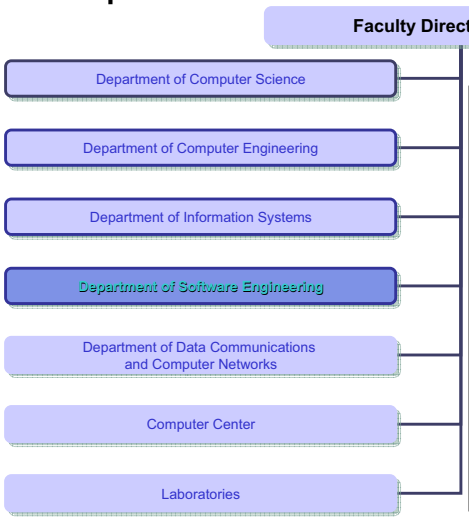
Departments'research areas



Department of Information Systems

- Data bases and management systems,
- Information system analysis and design,
- Artificial intelligent,
- Fuzzy systems.
- Grid Computing

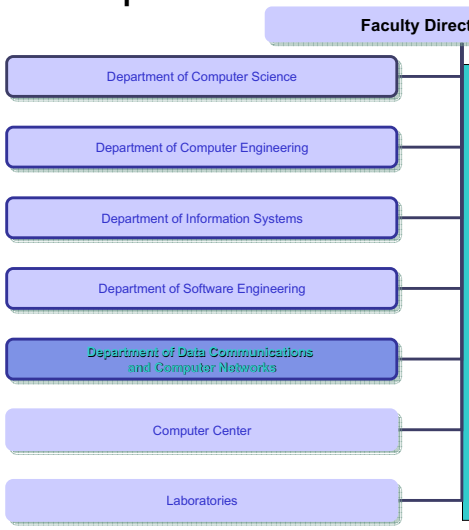
Departments'research areas



Department of Software Engineering

- Software engineering: methods, techniques, tools, standards,… for constructing software,
- Advanced programming models,
- Computer graphic, virtual reality
- Open Sources Software

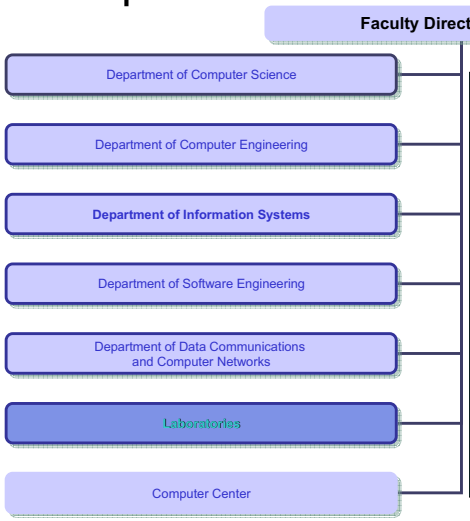
Departments'research areas



Department of Data Communications and Computer Networks

- Network Technologies
- QoS and Performance, Evaluation
- Distributed Systems
- Digital Communication,
- Multimedia Communication
  Image-Video processing
- Wireless Network and mobile communication
- Network Security and Biometric Based Information Security System

Departments'research areas



Laboratories

- Laboratory of Computer Systems and Information Processing,
- Laboratory of Internetworking, Data Communication and Multimedia.
- Laboratory of Software Engineering
- Laboratory of Information Systems
- Laboratory of Computer Science

Content

- **Introduction of Faculty of Information Technology (FIT)**
- **Training activities at FIT**
- **Improving the quality of training**
- **Research Activities at FIT**
- **Improving the level of collaboration in research with industry**
- **Proposals and Conclusion**

Training activities

Undergraduate programs

- Program for IT Engineering training: 5 years
  - Regular program in 5 specializations
  - IT Talented students training Program
  - Francophone Informatics Program
  - PFIEV Program
  - IT Vietnam-Japan Training Program
- International Training Program (ITP).
- Program for IT Engineering IT training: 2 years
- Program for IT Bachelor IT training: 3 years
- Courses for demands of the companies and organizations in Vietnam.

Training activities

- Five-year of training course for engineer's degree
  - 550 IT engineers per year.
    - 40-50 with aspiration to continue in Master/Phd. Courses in abroad
    - Others (80-90%) have a good job and career
- Two-year of engineering training course for those who have obtained an other university's degree
  - 50 IT engineers per year.
- Three year of training course for bachelor's degree
  - 300 IT bachelors per year.

Training activities

- **Post-graduate programs**
  - Two-years master program
    - Master of Information Technology.
      - 45 to 60 MSc per year (5-10 graduate with scientific paper in national conferences).
    - Master of Information Processing and Communication
      - 25 to 35 MSc per year (3-5 graduate with scientific paper in national conferences).
  - Ph.D program: about 5 per year

Training activities

- Objective:
  - Fundamental Knowledge
  - Technological Skill
  - English
  - High Level of Soft-Skill
- To meet demands of students:
  - Successful finding a good job in industry: 85-90%
  - Successful applying master and Phd. scholarship in abroad (5-8%) or continuing at master courses in Vietnam (4-5%)
  - Starting own business: establishing the private company, joint-stock company, …

Content

- **Introduction of Faculty of Information Technology (FIT)**
- **Training activities at FIT**
- **Improving the quality of training**
- **Research Activities at FIT**
- **Improving the level of collaboration in research with industry**
- **Proposals and Conclusion**

Improving the quality of training

- To improving the level of obtaining the Fundamental Knowledge
  - Curriculum Development:
    - Follow the ACM/IEEE Guideline (www.acm.org)
    - Reviewing the Teaching Materials for each subject by the end of academic year
  - Improving teaching process: to follow the in-process evaluation - homework, assignment, midterm, final exam
  - Starting the teaching assistant and student-assistant system

Improving the quality of training

- To improving the Technological Skill for students
  - Provide courses on technology know-how based popular companies such as Microsoft, IBM, Oracle, Cisco, Sun, …
  - Support FIT student in getting technological certifications:
    - Technical Consulting for technological certificate preparation
    - Marketing promotion program for discounted exam pricing
  - Organize students club for exchange the skill in technology: IBM Student club, Microsoft student club
  - Organize workshop/seminar on new technology/application
  - FIT has strong collaboration with: IBM, Microsoft, Oracle, ..

Improving the quality of training

- To improving the English and Soft-Skill
  - Increasing the students awareness of using English
  - Inviting the expert from industry or professor from foreign institutions to give presentation at FIT
  - Regularly organize the TOEIC or TOELF examination for FIT Students to check the level of English
  - Visiting and internship at companies to observer the real situation at industry
  - Organize the seminar for job orientation with the companies
  - Recruitment workshop
  - Thesis development on-site at the companies, the thesis topic also provided by the companies

Content

- **Introduction of Faculty of Information Technology (FIT)**
- **Training activities at FIT**
- **Improving the quality of training**
- **Research Activities at FIT**
- **Improving the level of collaboration in research with industry**
- **Proposals and Conclusion**

Research and Development

- Implement many science projects, including HUT level, region level, national level projects and projects with international collaboration
- Organize and co-organize the workshops, conferences and symposiums

R&D Projects 2002- 5/2008

- National Project of Science -Technology: 3
- National Project of Fundamentals: 16
- Project on Ministre level : 20
- Projects on HUT level : 23
- International Collaboration Projects 2002-2008

International Collaboration Projects

- Research Project: Advanced open Source Web Service Platform and Applications in HUT, 2004 – 2005, Collaboration VLIR-HUT (Belgium)
- Project: Linux & Open Source (C3LD) 2004 – 2006, Collaboration: AUF/Programme TIC/Formation/C3LD/010
- Project: Web Services, Collaboration with Japan
- Research Project: Vietnamese Speaker Recognition, Asia Institute, 2005-2007.
- JEAGAL Project: Joint European-Asian education and application development program on GALileo, 2005 – 2007, European (EC)

International Collaboration Projects

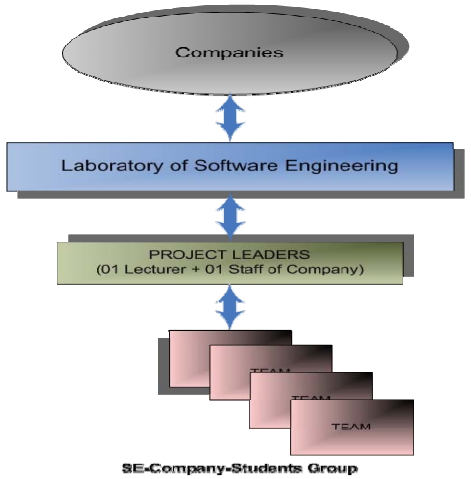
- Asia Link Project: Computational Logic and Logical Foundation of Computer Science, 2005-2007.
- Research project: BKGrid - an Open Source Platform for Grid and High Performance Computing, 2006-2007, Collaboration VLIR-HUT (Belgium)
- Research project: A Natural Language Interface for Querying Database and Automatically Generating Reports, 2006-2007, Collaboration VLIR-HUT (Belgium)
- Research project: Bio-PKI Based Information Security System, 2006-2008, Collaboration with MMU Malaysia
- A lot of the other cooperation activities with the International companies

Content

- **Introduction of Faculty of Information Technology (FIT)**
- **Training activities at FIT**
- **Improving the quality of training**
- **Research Activities at FIT**
- **Improving the level of collaboration in research with industry**
- **Proposals and Conclusion**

Improving the level of collaboration in research with industry

- Research agreement by theme with some companies
  - Regularly progress checking and monitoring
  - Using English for communication
- Onsite practice of students in some companies
  - Topic given by companies
  - Regularly evaluation and monitoring system



Improving the level of collaboration in research with industry

- 5-6 agreement per academic year
- 40-50 students (10%) involve in the practice (on-site) at the companies

Content

Proposals and Conclusion

- **Collaborative Training and Internship**
  - Support FIT students in practice and internship programs at the companies
  - Finance support FIT students in organizing the activities to improve the English and soft-skills such as seminars, workshops, …
  - Creating Innovation club to exchange the technological knowledge and skill between the students and experts from companies
  - Supporting FIT students in getting the technological certifications

Proposals and Conclusion

- **Collaborative Research**
  - Support young faculty members and students to research on new technologies and publish papers or participate international conference.
  - Facilitate student-led (engineer, master and PhD) and young faculty members research projects in FIT via partnership program with leading technological companies in the world
  - Transfer research results into improved systems and products and possibility to provide in the market.
- Conclusion: The stronger collaboration between the academia and industry is right way to enhance the education quality

Asia-Pacific
Economic Cooperation

2008/SMEWG/SYM/018
Agenda Item: 8.3

Software Outsourcing Human Resource: the Teams as Good as the Weakest Link

Purpose: Information
Submitted by: Vietnam

APEC Symposium on Improving Market Access for ICT Outsource SMEs
Hanoi, Vietnam
27–29 October 2008

The Team is as strong as the weakest link!

Toward a successful software engineering team

Nguyen The Trung
DTT Group
Acknowledged: Pro. John Vu, Boeing, CMU

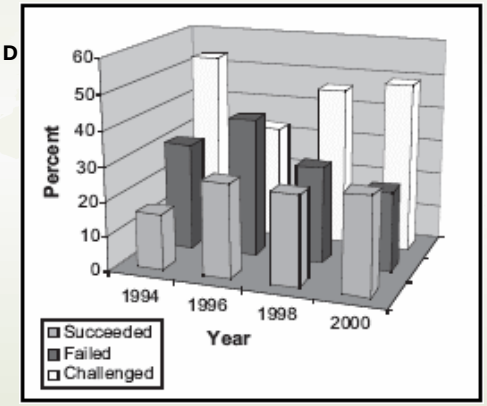
Software problems - No silver bullet

- " Software project, at least as seen by the non technical manager is usually innocent and straightforward, but is capable of becoming a monster of missed schedules, blown budgets, and flawed products".
  - **Complexity** : essential complexity and its nonlinear increases with size.

    a scaling-up of a software entity is not merely a repetition of the same elements in larger sizes, it is necessarily an increase in the number of different elements. In most cases, the elements interact with each other in some nonlinear fashion, and the complexity of the whole increases much more than linearly.
  - **Conformity** : much complexity comes from conformation to other interfaces; this complexity cannot be simplified out by any redesign of the software alone.
  - **Changeability** : the software product is embedded in a cultural matrix of applications, users, laws, and machine vehicles. These all change continually, and their changes inexorably force change upon the software product.
  - **Invisibility** : Software is invisible and unvisualizable

Why Software Projects fail

- **Unrealistic Schedules**
- **Inappropriate Staffing**
- **Changing Requirements D Development**
- **Poor-Quality Work**
- **Believing in Magic**



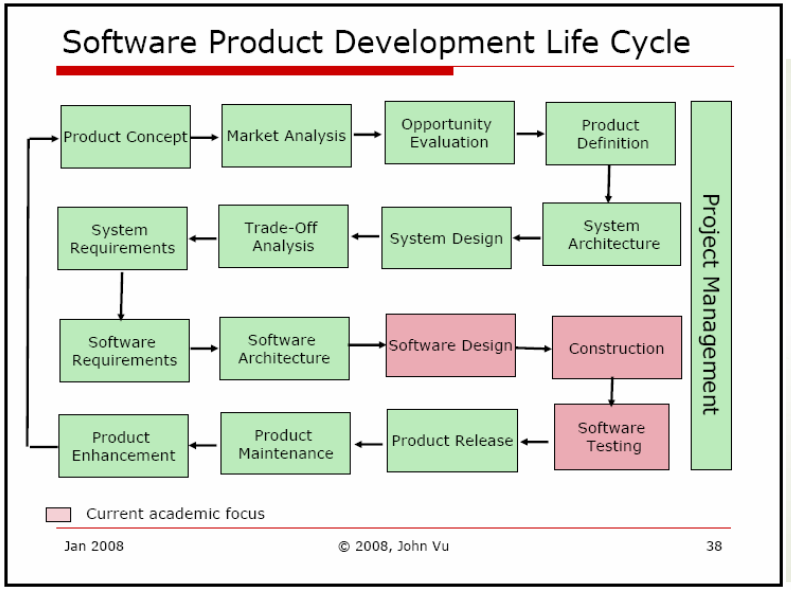
Five reasons why software projects fail, C Watts S. Humphrey , ComputerWorld

Challenges of Offshore Outsourcing

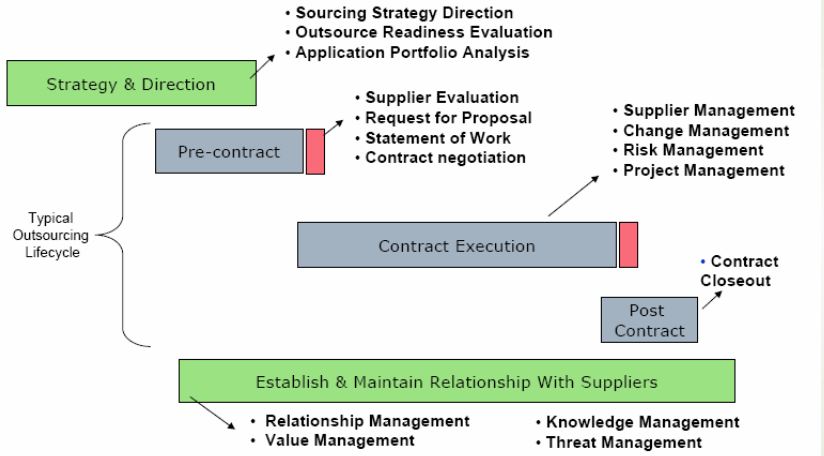
Challenges	%
Managing Communication	67%
Cultural Differences	51%
Lack of internal Processes for specifying work	40%
Lack of internal customer management skills	32%

www.cio.com

Software Product Development Life Cycle



Outsourcing lifecycle

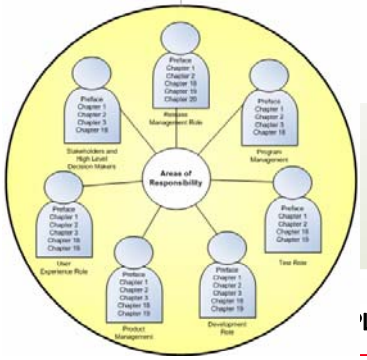


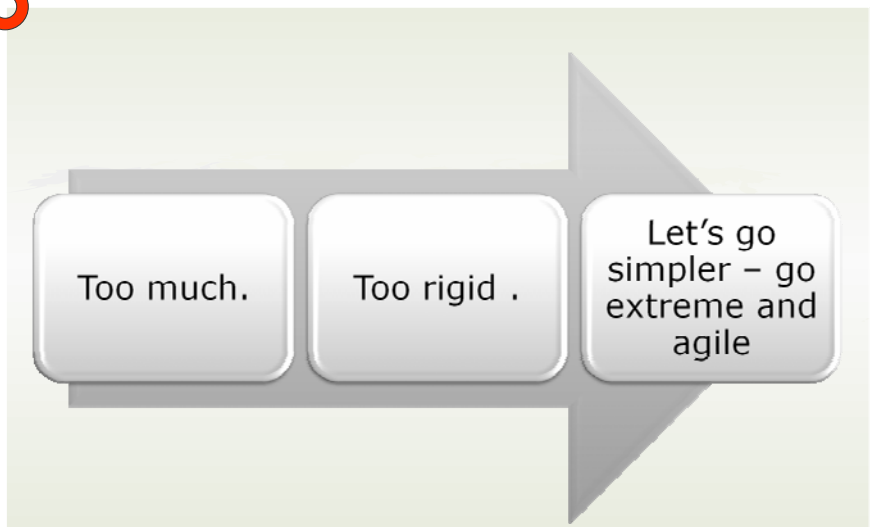
There are many roles in any SD process model !!!

Breadth and depth roles in RUP disciplines

Discipline	Breadth role	Depth role
Business Modeling	Business Process Analyst	Business Designer
Requirements	Systems Analyst	Requirements Specifier
Analysis and Design	Software Architect	Designer
Implementation	Integrator	Implementer
Test	Test Manager	Test Designer
	Test Analyst	Tester
	Test Designer	
Deployment	Deployment Manager	Tech Writer, Course Developer, Graphic Artist
Project Management	Project Manager	Project Manager
Environment	Process Engineer	Tool Specialist
Configuration and Change Mgt	Configuration Manager	Configuration Manager
	Change Control Manager	Change Control Manager

MSF Team model





Four roles in Scrum:

- The **scrum master** reviews the team's progress team and ensures time estimations are
- updated.
- The **product owner** writes user stories and defines acceptance tests.
- The **scrum team** estimates task durations and develops stories and unit tests.
- The **manager** provides directions to keep the work going according to plan and removes obstacles.

Six [core] roles in Feature-Driven Development (FDD):

- The **project manager** leads the team and reports on its progress.
- The **chief architect** is responsible for system design.
- The **development manager** is responsible for the development activities.
- The **chief programmers** provide technical leadership to the smaller teams.
- The **class owners** are developers who each own one class and are responsible for making all changes in it.
- The **domain experts** are the users.

Six roles in Lean Development:

- The **customer** provides the requirements.
- The **master developer** is responsible for system design.
- The **expertise leader** is responsible for specific technical areas such as GUI design, database development, and security.
- The **project leader** is responsible for time estimations and the team's progress.
- The **observer** takes notes on the team's process.
- The other team members are the **programmers**.

Six roles are mentioned in ASD Adaptive Software Development (ASD)

- The **executive sponsor** is responsible for the product being developed.
- The **developer** and **customer** representatives.
- The **facilitator** plans and leads the development sessions.
- The **project manager is** responsible for product delivery.
- The **scribe** records requirements, agreements and decisions reached.

Seven roles in Extreme Programming (XP):

- The **programmer** analyzes, designs, tests, programs, and integrates.
- The **customer** tells and writes stories to be implemented and decides when they will be implemented.
- The **tester** uses the customer's viewpoint in order to determine which items most require verification.
- The **tracker** measures progress quantitatively, by comparing estimations with actual results.
- The **coach** is responsible for the process as a whole.
- The roles of **consultant** and **boss** are external and are filled by people from outside the team.

Eight roles in Crystal Clear:

- The **sponsor** provides the mission statement.
- The **senior designer** produces the system design.
- The **user** helps with use cases and screen drafts.
- The **designer-programmers (designers**) design, code and test.
- Four additional merged roles are identified in Crystal Clear, which means that they can come from the people filling the above-mentioned roles:
- The **business expert** can come from the sponsor, user, or senior designer.
- The **coordinator** can come from the senior designer and is responsible for the schedule and the release sequence.
- The **tester** can come from the designers and is responsible for test results and defect reports.
- The **writer** can come from the designers and is responsible for the user manual.

Eleven roles in Dynamic Systems Development Method (DSDM):

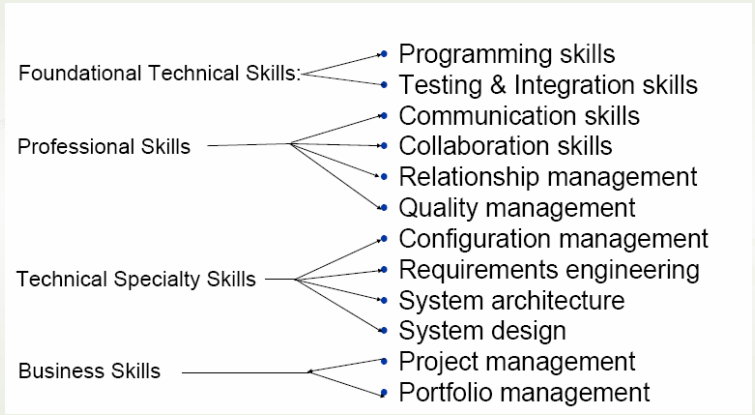
- The **executive sponsor** is a high-level executive who is responsible for the system and
- for its fast development progress.
- The **ambassador user** represents the entire user community.
- The **visionary user** makes sure that the vision of the product is not lost.
- The **advisor user** brings daily business knowledge to the development team.
- The **project manager** is responsible for ensuring project delivery, coordinating and reporting to the management.
- The **technical coordinator** reports to the project manager and assists all development teams.
- The **team leader** ensures that the team functions as a whole, and that the objectives are met.
- The **senior developer** interprets user requirements into prototypes and deliverable code.
- The **developer** assists with these tasks as part of DSDM skills development.
- The **facilitator** is responsible for managing the workshop process, an interactive communication technique for making decisions.
- The **scribe** records requirements, agreements and decisions reached.

PEOPLE ◦ SHARE ◦ TECHNOLOGY
DTT TECHNOLOGY GROUP

SOLUTION !!!

PEOPLE ◦ SHARE ◦ TECHNOLOGY
DTT TECHNOLOGY GROUP

Skills needed in outsourcing business



PEOPLE ◦ SHARE ◦ TECHNOLOGY
DTT TECHNOLOGY GROUP

Programming skills

Knowledge of implementation issues, including the use of virtual machines in language understanding; representation of data types; sequence control; data control, sharing, and type checking; run-time management; and language translation systems.

• Skills in the construction of the software components that are identified and described in the design documents. Including knowledge about translation of a design into an implementation language, program coding styles, software reuse, and the development and use of program documentation.

• Knowledge and skills in the translating a software design into an implementation programming language. Including knowledge about modular and incremental programming, structured programming, and knowledge of various programming paradigms (assembly, procedural, object-oriented, functional, and logic). It also includes knowledge about how to use source code development tools and programming language translation tools.

Testing and integration skills

• Ability to establish test cases and scripts that correct a defects in providing solution to a problem.

• Testing is a multi-stage process that consists of activities for validating

the software product, from the most primitive elements up to the fully integrated system.

• Ability to develop and conduct unit testing, performance testing, integration testing, system testing, and acceptance testing.

• Ability to verify and validate software components and final product to meet clients' requirements.

Requirements Engineering Skill …

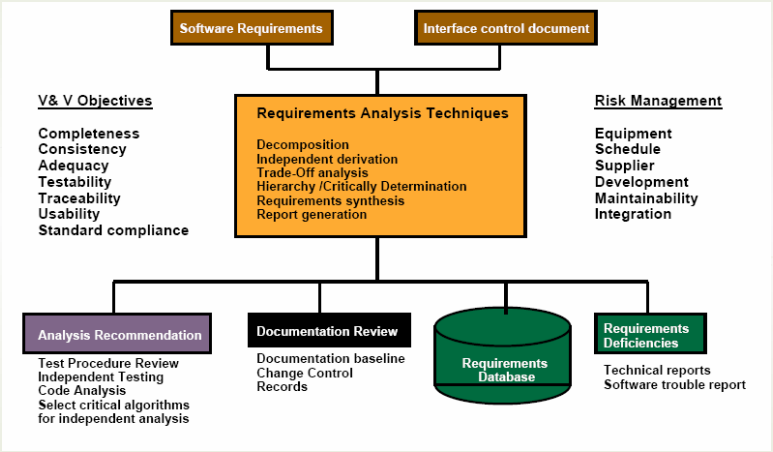
• Ability to work cooperatively with clients to obtain better requirements by understanding their business needs and to assist clients to write good requirements in their contract, to set up requirements baseline, and to manage requirements changes.

• Ability to obtain a precise formal requirements specification from the informal and often vague contracted work authorization written by clients.

• Ability to validate client's requirements and build traceability between requirements to system and software components.

• Ability to manage changes to requirements during contract duration and ensure formal configuration control of all changes against baseline.

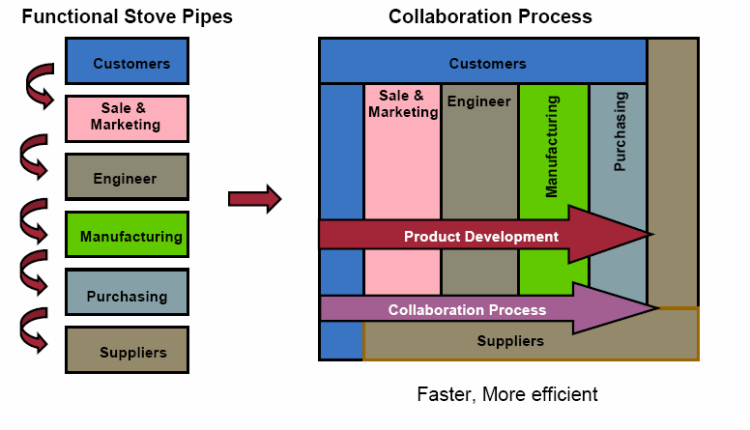
Requirements Analysis



Collaboration skills

• Ability to work together effectively as a team members and across job functions and levels to guarantee results in today's service oriented business environment.
• Ability to maximize the sharing of ideas, knowledge and technological know-how to make value-added decisions, resolve conflicts, and maintain trust.
• Ability to communication clearly and concisely to facilitate collaborative decision making skills.
• Ability to articulate vision, mission and objectives collaboratory.
• Ability to resolve conflict via a defined process
• Ability to maintain perspective in the workplace

Collaboration management



System Architecture

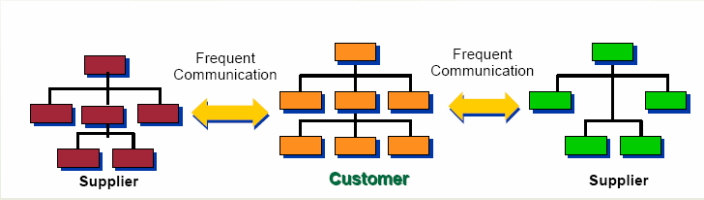
• Ability to design, understand, and evaluate software systems at an architectural level of abstraction.
• Knowledge of methods in planning, organizing, designing efficient, reliable computing systems to satisfy a high level, often vague requirements.
• Skills in architecting in both traditional Von-Neumann architecture and the evolution of non-Von Neumann architectures such as Pipelining; Reduced instruction set computer (RISC); Complex instruction set computer (CISC) architectures, multiprocessors and multi-computers, parallel programming, data flow architecture, interconnection networks, and neural networks.
• Skills in network-centric architectures: protocols and standards, transmission techniques and devices, speed and quality tradeoffs, and security and encoding algorithms.

System Design

• Ability to transform requirements into a description of how these requirements are to be implemented.
• Knowledge of variety of techniques and forms of representation to conduct architectural design, abstract specification, interface design, component design, data structure design, tasking design, and algorithm design.
• Skills in identifying and documenting the subsystems making up the overall system, and the relationships between and among the subsystems. Including knowledge about design methods and techniques for functional design, object-oriented design, real-time system design, and client-server system design.
• Skills and knowledge about the interface between subsystems and users including knowledge about interface design principles, task analysis and interface modeling, implementation tools, information presentation, design evaluation, and user documentation.

Communication

- • Suppliers must builds trust through frequent, open and honest communication with customers.
- • Meet and discuss when things are OK, Not just problems
- • Understand customers' expectations
- • Document the shared goals and expectations
- • Manage customer's stated project performance, measures and monitor them as a firm requirements
- • Track and resolve concerns and issues



Project Management

• Ability to define project objectives, assessing project needs and resources, developing estimates for the work to be performed, establishing the necessary commitments, and defining the plan for performing the work.

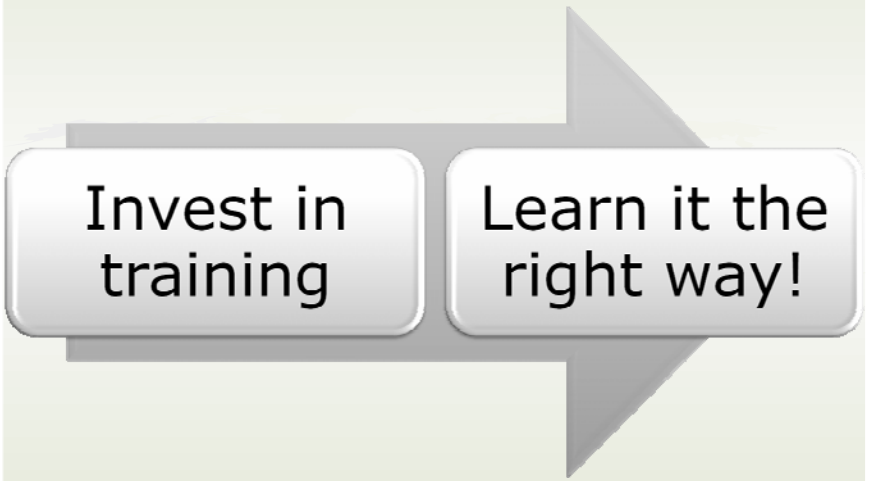
• Ability to prepare a project plan that include scope, goals and objectives, strategies, policy and estimates of size, functions, schedule, and resources needed

• Ability to manage and control project execution according to the project plan and metrics, managing changes and report status and capturing historical data.

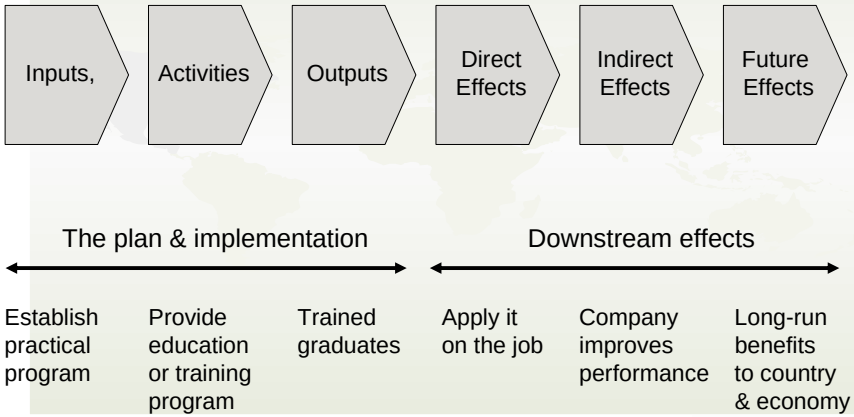
Relationship management

Ability to:

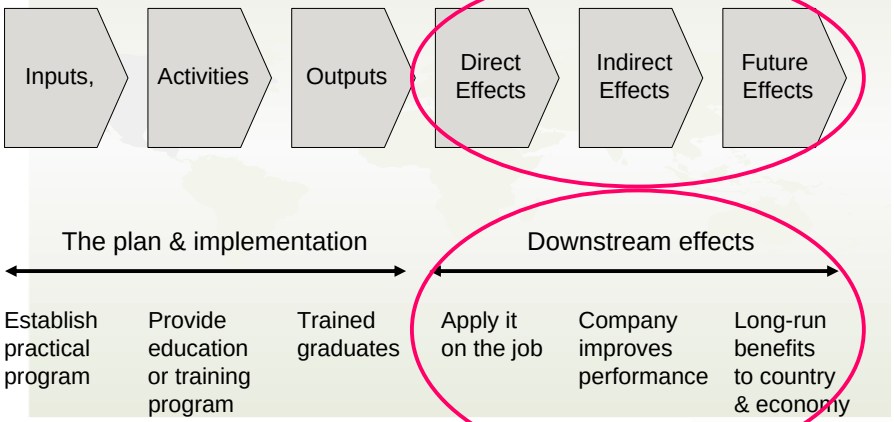
• Establishing trust
• Managing clients' expectations
• Ensuring positive experiences during interactions
• Managing cultural differences
• Ensuring confidentiality
• Managing relationships
• Preventing communication breakdowns
• Measuring service performance
• Developing employee satisfaction, avoid turnover
• Capturing lessons learned

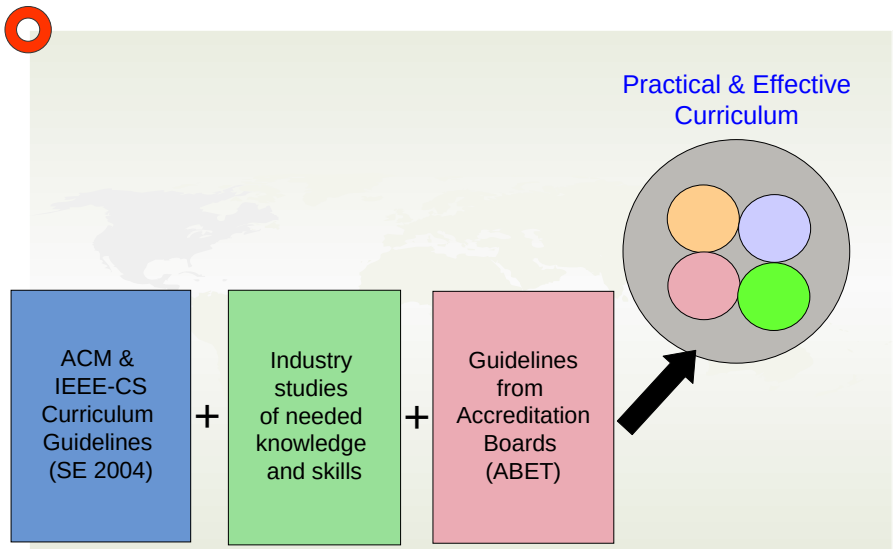


The Long Term Evaluation

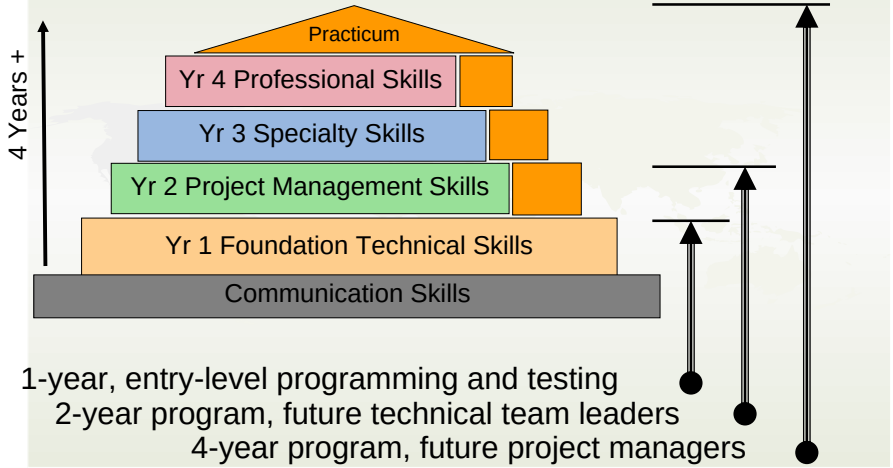


The Long Term Evaluation





A Practical 4-year Curriculum



Practical Software Engineering	ABET & Accreditation Criteria
Professional Practice	
Capstone project (2 semesters)	Demonstrate Mastery of Knowledge & Skills
Professional Seminars (Twice a year)	
Computing Fundamentals	
Fundamentals of Computing 1	Data Structures, JAVA, Algorithms
Fundamentals of Computing 2	Data Structures, OOD, UML, JAVA
Introduction to Network & Telecom	Distributed Programming, JAVA
Advanced Concepts in Computing	C#, OOD, Modeling, Problem Solving
Software Construction	Formal methods - PSP/TSP Programming
Information Systems Applications	Introduction to Databases, Data mining
Application Development Practices	Problem Solving
System Integration Practices	Problem Solving
Software Engineering	
Introduction to Software Engineering	Software Principles & Life Cycle Concepts
Software Architecture & Design	Concepts of Design & Tests
Software Testing	Concepts of Verification & Validation
Requirements Engineering	Concepts of Requirements Analysis
Software Project Management	Concepts of Software Management
Software Process & Quality	Concepts of Quality Management
Software Measurement & Analysis	Concepts of Measurements, Logic
Software Reuse & Integration	Concepts of Components & Integration
Group Dynamics & Communication	"Basic Communication & "Soft Skills"

US Industry Skills Recommendations	ABET & SE 2004 Student Outcomes	Practical Software Engineering Courses
Ability to apply knowledge of science, engineering and mathematics	Show mastery of the software engineering knowledge, and professional skill necessary to begin practice as a software engineer	Capstone project for Software Engineering Application Development Practices System Integration Practices Software Reuse & Integration
Ability to design and conduct experiments (analyze & interpret the data)		Requirements Engineering Advanced concept in Computing Software constructions Software Architecture & Design
Ability to function in a team	Work as an individual and as part of a team to develop and deliver quality software artifacts	Group Dynamics & Communication Applications Development Practices System Integration Practices Capstone Project for Software Engineering
Ability to design systems, components, process to meet customer's needs within realistic constraints	Reconcile conflicting project objectives, finding acceptable compromises within limitations of cost, time and knowledge in existing systems and organizations	Requirements Engineering Software Project management Software Architecture & Design Software Construction Software Testing Software Measurement & Metrics Software Reuse & Integration
Understand professional & ethical responsibility	Design appropriate solutions in one or more application domain using software engineering approaches that integrate ethical, social, legal and economic concerns	Introduction to Software Engineering Group Dynamics & Communication System Integration Practices

US Industry Skills Recommendations	ABET & SE 2004 Student Outcomes	Practical Software Engineering Courses
Ability to use techniques, knowledge, and skills to solve problems		Fundamental of Computing 1 Fundamental of Computing 2 Advanced Concept of Computing Software Architecture & Design Software Construction Applications Development Practices System Integration Practices Capstone Project For SE
Ability to communicate effectively	Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for problem identification and analysis, software design, development, implementation and documentation	Group Dynamics & Communication Requirements Engineering Software Project Management Software Architecture & Design
Knowledge of contemporary issues	Learn new models, techniques and technologies as they emerge and appreciate the necessity of continuing professional development	Introduction to Software Engineering Software Process & Quality Management Software Reuse and integration Introduction to Network & Communication
Understand the impact of an engineering solution in global economics, and the environmental/social context		Introduction to Software Engineering Introduction to Network Communication
Ability to work in one or more significant application domains		Introduction to Network Communication Information System Applications

Thank you