MAPPING OF THE SPECIFIC COMPETENCIES OF PROFILES ASSESSED BY CONAIC WITH THE ACM AND IEEE OBJECTIVES AND CONAIC AREAS OF KNOWLEDGE

This document has the purpose of mapping the IEEE, ACM and AIS curriculum guidelines body of knowledge objectives with the professional profile and specific competencies assessed by CONAIC. These include the Software Engineering IEEE and ACM (Ardis et. al 2015), Computer Science IEEE and ACM (Force 2013), Computer Engineering ACM and IEEE (Lyon 2004) and Information Systems AIS and ACEM (Topi et. al 2010) curriculum guidelines for undergraduate degree programs. This document also has the intention of mapping these same profiles and competencies to the CONAIC areas of knowledge that are included in the CONAIC Application for provisional membership in the Soul Accord.

Professional Profiles and Specific Competencies Assessed by CONAIC

The validation of the professional profiles assessed by CONAIC is shown below. As it can be seen, these profiles are those of Computing Professionals, with variations:

BA in Information Systems - A

Professional with knowledge and skills to improve organizational processes; exploit opportunities created by technological innovations; understand and resolve requests for information in organizations; design and manage the company architecture; identify and evaluate solutions, as well as their potential sources of supply or ways to be performed; manage data security and infrastructure, in addition to understand, manage and control IT risks; manage changes generated by IT solutions in organizations.

Professional with solid knowledge of Information Technologies applied to the administrative process of the organizations. Technological strategist that develops and implements solutions to support the competitiveness; he/she develops and implements solutions to support competitiveness and processes for managing knowledge and providing agility to organizations. This is an eminently professional profile type, but it does not exclude the need to pursue postgraduate studies, both in information sciences and technologies, and in areas benefiting from their contributions.

The bachelor's degree in Information Systems graduate will need to:

- Have proper knowledge of the internal and external organizational structures, as well as the importance of development of human resources.
- Visualize, describe and define the functioning and operation of a system and its interaction with the development of the organization.
- Perform feasibility analysis of economic, technological, social and human behavior for effective decision-making in organizations.
- Design, implement, monitor, and manage database systems for the administration of large volumes of information in organizations, optimizing hardware and software resources required, making use of the correct tools to support decision-making.

Profile A Specific Competencies:

ACM and AIS Body of Knowledge Areas for Information Systems	No.	Competency	Attribute	CONAIC Areas of knowledge
Information Systems Fundamentals **2.0 Organizational and Management Concepts** 2.1 General Organization Theory 2.2 Information Systems Management 2.3 Decision Theory 2.4 Organizational Behavior 2.7 Managing the Process of Change 2.8 Legal and Ethical Aspects of IS 2.9 Professionalism 2.10 Interpersonal Skills	**1**	Improves Organizational Processes	It uses Administrative Theory principles, administrative process and the organization's characteristics. Recognizes the context of the organization (public and private). Applies the theories of administration (Business Process Management) to improve the performance of organizations	SS6 Types and basic principles of organizations SS7 Admistrative Procedures ES11 Basic Accounting Principles ES12 Cost Accounting. Catalogs accounts ES13 Financial planning ES14 Budgets ES15 Tax issues in organizations ES16 basics of microeconomics SS17 Basics of macroeconomics ES18 Business Economics ES31 legal considerations ES32 Commercial law ES35 Ethics ES36 Authorship information systems EN37 Impact of technology ES38 social impact of

				computing
Information Systems Development **3.0 Theory and Development of Systems** 3.1 Systems and Information Concepts 3.2 Approaches to Systems Development 3.8 Information and Business Analysis	**2**	Proposes and evaluates projects on information technologies	Recognizes Management needs in a system using techniques to identify, collect, analyze, prioritize, document, verify and validate requirements in the environment and processes in the development of computer systems	ES19 computer units in organizations ES1 Megatrends, globalization and convergence
	3	Establishes mechanisms for computer auditing	Creates or proposes methods and strategies for carrying out audits (documentation and monitoring of standards). Applies Computer Audits	ES30 Computer audit ES33 Policy computer ES34 Human rights.
Information Systems Development **3.0 Theory and Development of Systems** 3.2 Approaches to Systems Development 3.3 Systems Development Concepts and Methodologies 3.4 Systems Development Tools and Techniques	**4**	Develops solutions in several application domains, making use of software engineering principles and methods	Capacity to analyze, design and build applications in desktop, mobile or network environments, choosing the appropriate paradigm and programming languages.	TI26 Business Applications
Information Technology **1.0 Information**	**5**	Describes database basic	Describes concepts such as Database, Database Management	TI1 General Concepts TI2 The hierarchical

Technology 1.6 Database		concepts	System, Relational, Hierarchical and Network Database.	model TI3 The network model TI4 The relational model TI5 Entity-relationship model TI6 Relational design TI7 Alternative models TI8 Distributed databases TI9 Handlers TI10 Query languages
Information Technology **1.0 Information Technology** 1.6 Database	**6**	Manages Database Systems	Manages access control and user authorization, in addition to optimizing hardware and software resources needed for these systems. Develops routines and triggers to automate some tasks of the system itself and of the databases. Describes and applies database concepts to improve performance and provide greater security, as well as datawarehouse, OLAP, Data Mining, Big Data, Data Analytics.	TI11 Architecture of database systems TI12 Basics IT13 File Handling TI14 Safety databases
Information Systems Theory and Practice **3.0 Theory and Development of Systems** 3.1 Systems and Information Concepts 3.2 Approaches to Systems	**7**	Employs the Software Industry Best Practices	Describes and uses Best Practices concepts in the provision of Information Technologies Services based on International Quality Standards	ES21 Tenders Companies in real time ES27 Control and monitoring of business processes ES2 organizational Socialization, interaction distance and distance

Development 3.3 Systems Development Concepts and Methodologies 3.4 Systems Development Tools and Techniques				communication processes. ES3 Social changes for Internet use. ES4 the digital gap (digital ES5 Quality Models (CMM, ISO, ITIL, MOPROSOFT, 6SIGMA, among others) from the organizational approach TI25 Organization and administration
Information Technology **1.0 Information Technology** 1.1 Computer Architectures 1.2 Algorithms and Data Structures 1.3 Programming Languages 1.4 Operating Systems 1.5 Telecommunications 1.7 Artificial Intelligence	**8**	To determine the appropriate hardware and software platforms.	To define, evaluate and select relevant hardware and software platforms to enable the development and implementation of systems, services and applications.	ES22 Administration computer units ES23 Real computer TI28 Distributed systems against computer network
Information Systems Theory and Practice **2.0 Organizational and Management Concepts** 2.7 Managing the Process of Change	**9**	Designs the strategic plan using IT	Uses different planning processes and computer projects in order to align IT area's objectives with Organization's objectives, so as to achieve a sustainable value generation.	SS9 Competitiveness in the digital age ES26 Promoting sustainable development

3.0 Theory and Development of Systems 3.9 Information Systems Design	**10**	Proposes Comprehensive Computer Solutions	Understands the needs of the organization, Information Systems, modeling and design of Information Systems, organizational processes	TI19 Systems approach TI20 System types TI22 Analysis and design
Information Systems Development **3.0 Theory and Development of Systems** 3.8 Information and Business Analysis 3.9 Information Systems Design 3.10 Systems Implementation and Testing Strategies 3.12 Systems Development for Specific Types of Information Systems	**11**	Builds business applications	Develops new business models that take advantage of technological innovations. Analyzes, develops and builds information systems in order to model situations of the real environment, solve problems, and optimize decision-making to support the organization's Management.	ES25 Management Innovation and Technology TI27 Enterprise service-oriented architectures (SOA).
Information Systems Development **3.0 Theory and Development of Systems** 3.6 Risk Management 3.7 Project Management	**12**	IT Project Management	Applies a systematic methodology to initiate, plan, execute, monitor and close IT projects; manages the team, communication, calendar, resources, risks, and quality of the project.	ES10 Evaluation of information technology projects ES20 Business and IT projects ES28 Administration IT staff. ES29 Update IT staff

Bachelor's degree in Software Engineering - B

Specialist in the production of quality software systems for the solution of various problems in the environment. Responsible for the formulation, planning, implementation and maintenance of information systems that ensure the availability of high service levels.

Must have a solid training in analysis techniques and information systems design, and in the setup of computing services environments and networks, as well as the knowledge of programming tools and software engineering, in order to build programs and application systems with characteristics of finished and competitive products.

The bachelor's degree in Software Engineering graduate will need to:

- Describe main software concepts and models for use in systems development.
- Identify and analyze problems in order to propose, design, build, verify, and document software solutions.
- Apply techniques and methodologies for the production of quality software.
- Promote collaborative work and responsibility in the resolution of problems for the development of the competencies required.

Profile B Specific Competences:

Software Engineering ACM/ IEEE Body of Knowlede objectives	No.	Competency	Attribute	Areas of knowledge
REQ Requirements analysis and specification • REQ.rfd Requirements fundamentals • REQ.er Eliciting requirements • REQ.rsd Requirements specification and documentation • REQ.rv Requirements validation	**1**	Performs software engineering requirements	Recognizes the context and needs, and individuals involved in a system using techniques to identify, collect, analyze, prioritize, document, verify and validate the requirements in the context of life cycles and software development processes.	PI25 General Concepts about systems PI26 System analysis PI27 Methodologies for analysis
DES Software design • DES.con Design concepts	**2**	Designs Software	Designs and Evaluates the behavior, architecture and	PI28 system design PI29 design methodologies

• DES.str Design strategies • DES.ar Architectural design • DES.hci Human-computer interaction design • DES.dd Detailed design • DES.ev Design evaluation			interface of software solutions based on requirements and using strategies, methods, techniques and modeling languages characteristic to software design.	PI30 Service-oriented architecture
CMP Computing essentials • CMP.cf Computer science foundations • CMP.ct Construction technologies • CMP.tl Construction tools	**3**	Builds software	Develops software for different types of applications, using programming methodologies and paradigms in the context of life cycles and software development processes, with the required quality attributes.	PI41 Evolution software development PI42 evolution of the software industry in Mexico PI43 Characteristics of industrial software products
VAV Software verification and validation • VAV.fnd V&V terminology and foundations • VAV.rev Reviews and static analysis • VAV.tst Testing	**4**	Carries out Software Testing	Plans, assigns and runs types, techniques, processes and controls inside test scenarios according to the required quality attributes.	PI37 Rules, standards and documentation
PRO Software process • PRO.cm Software configuration management • PRO.evo Evolution processes and activities	**5**	Carries out Software Maintenance	Applies maintenance types, processes and techniques, in accordance with the required quality attributes.	PI31 Lifecycle systems PI36 Maintenance and extensions
PRO Software process • PRO.pp Project planning and tracking	**6**	Manages software projects	Uses methods, strategies, processes, tools and techniques for software projects	PI33 Agile design methodologies PI34 Control the

•			management.	progress of software projects
3 QUA Software quality • QUA.cc Software quality concepts and culture • QUA.pca Process assurance • QUA.pda Product assurance	**7**	Estimates parameters for the software project	Applies metrics for software estimation (size, cost, effort, personnel, time, productivity, quality and documentation) in accordance with system life cycle models.	PI32 Estimate. PI39 Quantitative measurement of quality
QUA Software quality • QUA.cc Software quality concepts and culture • QUA.pca Process assurance • QUA.pda Product assurance	**8**	Ensures Software Quality	Uses techniques, tools, and strategies for planning, ensuring and controlling a software product quality.	PI38 Software quality control PI39 Quantitative measurement of quality
SEC Security • SEC.sfd Security fundamentals • SEC.net Computer and network security • SEC.dev Developing secure software 8	**9**	Establishes security mechanisms	Creates, Evaluates or proposes methods and strategies to evaluate safety and selection of criteria to avoid security vulnerabilities in the software.	PI40 Audit systems SB18 virus treatment RE25 Integrity RE26 Security RE27 Reliability
MAA Software modeling and analysis • MAA.md Modeling foundations • MAA.tm Types of models • MAA.af Analysis fundamentals	**10**	Uses life cycles	Uses elements and criteria in the use of life cycle models in accordance with the context of software development processes.	PI41 Evolution software development PI42 evolution of the software industry in Mexico
VAV Software verification and validation • VAV.fnd V&V	**11**	Checks software solutions	Uses various test models in order to ensure software	PI35 Test methods

terminology and foundations • VAV.rev Reviews and static analysis • VAV.tst Testing • VAV.par Problem analysis and reporting		quality	product quality.	
CMP Computing essentials • CMP.cf Computer science foundations • CMP.ct Construction technologies • CMP.tl Construction tolos	**12**	Uses software creation tools	Used industrial methods and CASE tools for the different stages in the software process.	PI44 Industrial methods for creating software PI45 Tools for creating software

Bachelor's degree in Computer Science - C

Professional with the ability and skills required for computer science study and development, resulting in decision making and planning in innovative applications within baseline software areas, the principles that characterize formal sciences, and the design and building of complex reality models, taking care of their consistency, efficiency and performance.

The bachelor's degree in Computer Science graduate will need to:

- Apply the knowledge acquired in computer science.
- Create environments, facilities and innovative computer applications within different environments and provide efficient solutions.
- Build baseline software and applications.
- Know how to take decisions and planning in innovative applications.
- Application of exact sciences in order to theoretically and practically develop complex reality models.
- Design, plan, and generate environments, models and projects, making use of the New Information Technologies.
- Implement and evaluate computer systems in various contexts to promote safety and sustainability.
- Know how to install, monitor, and tune systems in operational environments (databases, operating systems, middleware software, among others).

Profile C Specific Competences:

ACM/IEEE Body of Knowledge for Computer Science	No.	Competency	Attribute	CONAIC Areas of knowledge
SE-Software Engineering CN-Computational Science	**1**	Approaches and solves mathematical problems	Recognizes the context and needs, and individuals involved in a system using techniques to identify, collect, analyze, prioritize, document, verify and validate requirements.	PI1 History of Computing PI2 Algorithmic basic PI3 structured approach MA26 propositional logic MA27 predicate logic MA28 sets
DS-Discrete Structures AL-Algorithms and Complexity	**2**	Makes representations of mathematical entities (objects and situations).	Uses methods, strategies, processes, tools and mathematic techniques for the representation of objects and scenarios.	PI4 Approach objects PI6 static structures in main memory PI7 dynamic structures in main memory PI8 secondary structures in memory PI9 File Organization PI10 Classification PI11 search PI12. Prediction
AL-Algorithms and Complexity PBD-Platform-based Development	**3**	Builds algorithms and software.	Construction of algorithms and quality software____through methodologies and programming languages in order to provide efficient solution to problems.	MA33 finite state machines MA34 recognition languages MA35 Formal languages
IS-Intelligent Systems	**4**	Uses artificial intelligence methods and approaches.	Use of artificial intelligence methods and approaches, and pattern recognition for problem solving, with advanced methods	IH4 Logic and problem solving IH5 Search

			and techniques.	IH6 Special languages IH7Learning IH8 Deduction IH9 Neural Networks IH10 Characterization of expert systems IH11 Reasoning and control IH12 Vision IH13 Robotics IH14 elements for the syntactic and semantic process
SDF-Software Development Fundamentals PL-Programming Languages AR-Architecture and Organization OS-Operating Systems	**5**	Implements problem-solving software through computer approaches.	Development and implementation of problem-solving software by using the appropriate programming language, operating system, and architectures.	IH2 Programming Techniques IH3 products PI16 imperative programming PI17 Object-oriented programming PI18 Functional programming PI19 Programming logic PI21 comparison of languages MA38 Complexity MA39 Decidability
CN-Computational	**6**	Identifies, models and implements solutions	Identification, modeling, evaluation and implementation	R28 distributed

Science SF-Systems Fundamentals		through computer science	of an efficient solution to a real problem through computer science techniques.	computing R29 Mobile Computing R30 Programming Internet. HTML and XHTML R31 Middleware R32 Web Services IH15 development process IH16 interaction styles IH17 Design principles MA36 Turing machines MA37 Recursive functions
CN-Computational Science	**7**	Apply the scientific method to computer science problems.	Application of scientific method as a means to solve computer science problems that allow him/her to test hypotheses on algorithmic behaviors.	PI13 complexity measures PI14 analysis algorithms PI15 Strategies for constructing algorithms
SE-Software Engineering GV-Graphics and Visualization IAS-Information Assurance and	**8**	Produces programs by applying visual programming	Uses visual programming techniques, tools, and strategies for planning, securing and controlling a software product.	PI5 visual approach PI20 Visual programming and events

Security				

Computer Engineering - D

Professional with the mission of building, configuring, evaluating and selecting computer jobs and service environments, capable of generating new technology and finding and implementing efficient solutions in organizations. He/She will have command of the theoretical principles and practical and methodological aspects that support the design and development of complex systems, specification of hardware architectures and configuration of computer networks.

The Computational Engineering graduate will need to:

- Describe concepts, techniques and methodologies for the design and development of complex systems, hardware architectures and configuration of networks for use in the implementation of efficient computer solutions in organizations.
- Apply techniques, methodologies and mathematical models in the design, implementation, and evaluation of computer systems, as well as process automation through its components (microprocessors, circuit boards, routers and other embedded devices) to meet computing needs in organizations that use them.
- Promote collaborative work and responsibility in the resolution of problems for the development of the competencies required.
- Permanently incorporate new technologies to the needs of information while minimizing environmental impact.

Profile D Specific Competences:

ACM/IEEE Body of Knowledge Computer Engineering	Bloom Level	No.	Competency	Attribute	CONAIC areas of knowledge
CE-CAO Computer Architecture and Organization CE-CAO0 History and overview CE-CAO1 Fundamentals of computer architecture CE-CAO2 Computer arithmetic	**2**	**1**	Describes computer components and systems	Recognizes and identifies the internal components of computer systems.	AC18 History and evolution AC19 classical von Neumann architecture AC20 alternative

CE-CAO3 Memory system organization and architecture CE-CAO4 Interfacing and communication CE-CAO5 Device subsystems CE-CAO6 Processor systems design CE-CAO7 Organization of the CPU CE-CAO8 Performance CE-CAO9 Distributed system models CE-CAO10 Performance enhancements CE-CAO11 Crosscutting Issues					architectures AC21 Microprocessors SB8 History and evolution SB9 architecture of an operating system
CE-CSE Computer Systems Engineering CE-CSE0 History and overview CE-CSE1 Life cycle CE-CSE2 Requirements analysis and elicitation CE-CSE3 Specification CE-CSE4 Architectural design CE-CSE5 Testing CE-CSE6 Maintenance CE-CSE7 Project management CE-CSE8 Concurrent (hardware/software) design CE-CSE9 Implementation CE-CSE10 Specialist systems CE-CSE11 System-level test and diagnosis CE-CSE12 Reliability and fault tolerance CE-CSE13 Error detecting and correcting codes **CE-ALG Algorithms and Complexity** CE-ALG0 History and	**4,5**	**2**	Develops computer solutions.	Analyzes and builds real-world solutions based on mathematical models.	MA20 Basics. Concept and use of simulation MA23 Linear Programming MA24 nonlinear programming, integer and dynamic MA25 Network Analysis SB19 Chargers and linkers SB20 Management and monitoring and tools SB21 High performance

overview CE-ALG1 Basic algorithmic analysis CE-ALG2 Algorithmic strategies CE-ALG3 Computing algorithms CE-ALG4 Distributed algorithms CE-ALG5 Algorithmic complexity CE-ALG6 Basic computability theory					
CE-SPR Social and Professional Issues CE-SPR2 Methods and tools of analysis	**3**	**3**	Uses techniques, skills, and modern computer tools	Applies modern computer tools that facilitate reaching solutions to problems raised.	MA21 Special programming techniques and languages
CE-NWK Computer Networks CE-NWK0 History and overview CE-NWK1 Communications network architecture CE-NWK2 Communications network protocols CE-NWK3 Local, wide area, and wireless networks CE-NWK4 Client-server computing CE-NWK5 Data security and integrity [4] CE-NWK6 Wireless and mobile computing CE-NWK7 Performance evaluation CE-NWK8 Data communications CE-NWK9 Network management CE-NWK10 Compression and	**5**	**4**	Designs and implements networks of personal, local, and global computers	Develops secure connectivity solutions.	RE1 Theoretical concepts RE2 codes. Information coding RE3 errors RE4 Types and links RE5 Communications RE6 transmission modes RE7 Media and physical elements RE8 communication devices RE9 communications services

decompression					RE10 Telecommunications. RE11 The ISO/OSI model RE12 Architecture RE13 Standards and organizations RE14 Local networks (LAN) RE15 wide networks (WAN). RE16 Metropolitan Area Networks and Internet Access Subscriber RE17 low level protocols RE18 high-level protocols RE19 Protocols for multimedia networks RE20 Theory interconnections EM21 devices for interconnection RE22 network design elements RE23 interconnected networks RE24 Internet SB16 backup and recovery systems. SB17 Monitoring

CE-SWE Software Engineering CE-SWE0 History and overview CE-SWE1 Software processes CE-SWE2 Software requirements and specifications CE-SWE3 Software design CE-SWE4 Software testing and validation CE-SWE5 Software evolution CE-SWE6 Software tools and environments CE-SWE7 Language translation CE-SWE8 Software project management CE-SWE9 Software approaches and software fault tolerance	**4**	**5**	Analyzes the existing computer solutions in order to propose innovative solutions	Identifies viable, sustainable and innovative solutions to problems raised.	ES26 Promoting sustainable development EN37 Impact of technology **ES38 social impact of computing**
CE-CSG Circuits and Signals CE-CSY0 History and overview CE-CSY1 Electrical Quantities CE-CSY2 Resistive Circuits and Networks CE-CSY3 Reactive Circuits and Networks CE-CSY4 Frequency Response CE-CSY5 Sinusoïdal Analysis CE-CSY6 Convolution CE CSY7 Fourier Analysis CE CSY8 Filters CE-CSY9 Laplace Transforms **CE-DIG Digital Logic** CE-DIG0 History and overview CE-DIG1 Switching theory CE-DIG2 Combinational logic circuits CE-DIG3 Modular design of combinational circuits CE-DIG4 Memory elements CE-DIG5 Sequential logic circuits CE-DIG6 Digital systems design CE-DIG7 Modeling and simulation CE-DIG8 Formal verification CE-DIG9 Fault models and testing CE-DIG10 Design for testability **CE-ELE Electronics** CE-ELE0 History and overview CE-ELE1 Electronic properties of materials CE-ELE2 Diodes and diode circuits	**3**	**6**	Implements computer architectures.	Apply various computer architectures in order to implement comprehensive solutions in computational systems.	AC1 Mechanical AC2 Optics AC3 Modern physics AC4 Electricity AC5 Magnetism AC6 Electrical circuits AC7 principles of electronics AC8 electronic circuits AC9 integrated circuit technologies AC10 special purpose devices AC11 Number Systems and Codes AC12 combinational circuits AC13 Sequential Circuits

CE-ELE3 MOS transistors and biasing CE-ELE4 MOS logic families CE-ELE5 Bipolar transistors and logic families CE-ELE6 Design parameters and issues CE-ELE7 Storage elements CE-ELE8 Interfacing logic families and standard buses CE-ELE9 Operational amplifiers CE-ELE10 Circuit modeling and simulation CE-ELE11 Data conversion circuits CE-ELE12 Electronic voltage and current sources CE-ELE13 Transistor amplifier design CE-ELE14 Power circuits CE-ELE15 Feedback in electronics CE-ELE16 Active filters CE-ELE17 Integrated circuit building blocks **CE-OPS Operating Systems** CE-OPS0 History and overview CE-OPS1 Design principles CE-OPS2 Concurrency CE-OPS3 Scheduling and dispatch CE-OPS4 Memory management CE-OPS5 Device management CE-OPS6 Security and protection CE-OPS7 File systems CE-OPS8 System performance evaluation **CE-ESY Embedded Systems** CE-ESY0 History and overview CE-ESY1 Embedded microcontrollers CE-ESY2 Embedded programs CE-ESY3 Real-time operating systems CE-ESY4 Low-power computing CE-ESY5 Reliable system design CE-ESY6 Design methodologies CE-ESY7 Tool support CE-ESY8 Embedded multiprocessors CE-ESY9 Networked embedded systems CE-ESY10 Interfacing and mixed-signal systems					AC14 functional units AC15 Memories AC16 Storage Devices (Peripherals External storage devices). AC17 peripheral input-output devices AC22 computer equipment configurations AC23 Forms processing AC24 Client-server model AC25 Assembly Language AC26 specialized configurations AC27. Card design and printed circuit boards SB10 performance of an operating system SB11 handling devices and special services SB12 special types of operating systems SB1 Assemblers (Macroassembler). SB2 Macroensambladores SB3 Macroprocesadores (macroprocessor). SB4 Interpreters SB5 Compilers SB6 Topics compilation SB7 integrated environments
CE-ESY Embedded Systems CE-ESY4 Low-power computing	**3,6**	**7**	Proposes alternative	Optimizes energy	AC28 Buildings

CE-ESY5 Reliable system design			solutions that optimize the use of energy	consumption by applying knowledge and evaluating options in the design of solutions.	AC29 Power
CE-CSE Computer Systems Engineering CE-CSE4 Architectural design CE-CSE8 Concurrent (hardware/software) design CE-CSE9 Implementation	**5**	**8**	Proposes innovative solutions that meet the needs of computer systems in both software and hardware	Permanently incorporates new technologies to the needs of information.	PI22 Relations between algorithms and architectures PI23 concurrent algorithms PI24 Parallelism SB13 virtualization SB14 Hypervirtualization SB15 graphics environments

APPENDIX — ACM/IEEE BODY OF KNOWLEDGE

Table 1 IS Curriculum Areas

Curriculum Presentation Area	Description
Information Systems Fundamentals	Information systems fundamentals include a broad introduction to the field of Information Systems and information technology plus instruction designed to improve personal productivity in an organization through effective and efficient use of information technology.

Information Systems Theory and Practice	Students will be introduced to concepts and theories that explain or motivate methods and practices in the development and use of information systems in organizations. The concepts and theories will include systems, management, and organization, information, quality, and decision making. The relationship of information systems to corporate planning and strategy and concepts relating information technology to comparative advantage and productivity are explained. The concepts and practices underlying the use of information technology and systems in improving organizational performance are presented.
Information Technology	Students will gain breadth and depth in the technical aspects of the discipline. Computing system architectures, operating systems software, and interconnection of information resources through networking are major components of presentation and discussion. Students will be expected to develop significant skills by participating in installation, configuration, and operation of the technologies.
Information Systems Development	Students will work in teams to analyze problems and design and implement information systems. Systems analysis provides experience determining system requirements and developing a logical design. Instruction in physical design of information systems will ensure that the students can use a logical design to implement information systems in both a DBMS and in emerging development environments. Students should be exposed to a variety of development approaches.
Information Systems Deployment and Management	Students engage in numerous projects. Management of the information systems function, systems integration, and project management to ensure project quality are integral components of this curriculum area.

Body of Information Systems Knowledge

1.0 Information Technology

- 1.1 Computer Architectures
- 1.2 Algorithms and Data Structures
- 1.3 Programming Languages
- 1.4 Operating Systems
- 1.5 Telecommunications
- 1.6 Database
- 1.7 Artificial Intelligence

2.0 Organizational and Management Concepts

- 2.1 General Organization Theory
- 2.2 Information Systems Management
- 2.3 Decision Theory
- 2.4 Organizational Behavior
- 2.7 Managing the Process of Change
- 2.8 Legal and Ethical Aspects of IS
- 2.9 Professionalism
- 2.10 Interpersonal Skills

3.0 Theory and Development of Systems

- 3.1 Systems and Information Concepts
- 3.2 Approaches to Systems Development
- 3.3 Systems Development Concepts and Methodologies
- 3.4 Systems Development Tools and Techniques
- 3.5 Application Planning
- 3.6 Risk Management
- 3.7 Project Management
- 3.8 Information and Business Analysis
- 3.9 Information Systems Design
- 3.10 Systems Implementation and Testing Strategies
- 3.11 Systems Operation and Maintenance
- 3.12 Systems Development for Specific Types of Information Systems

Table 2. IS 2002 Body of Knowledge Presented as a Two Level Hierarchy

Table 3. Computer Engineering Body of Knowledge Areas

CE-ALG	Algorithms and Complexity
CE-CAO	Computer Architecture and Organization
CE-CSE	Computer Systems Engineering
CE-CSG	Circuits and Signals
CE-DBS	Database Systems
CE-DIG	Digital Logic
CE-DSP	Digital Signal Processing

CE-ELE	Electronics
CE-ESY	Embedded Systems
CE-HCI	Human-Computer Interaction
CE-NWK	Computer Networks
CE-OPS	Operating Systems
CE-PRF	Programming Fundamentals
CE-SPR	Social and Professional Issues
CE-SWE	Software Engineering
CE-VLS	VLSI Design and Fabrication
CE-DSC	Discrete Structures
CE-PRS	Probability and Statistics

Table 4 ACM and IEEE Software Engineering body of knowledge

KA/KU	Title	Hours		KA/KU	Title	Hours
CMP	**Computing essentials**	**152**		**DES**	**Software design**	**48**
CMP.cf	Computer science foundations	120		DES.con	Design concepts	3
CMP.ct	Construction technologies	20		DES.str	Design strategies	6
CMP.tl	Construction tools	12		DES.ar	Architectural design	12
				DES.hci	Human-computer interaction design	10
				DES.dd	Detailed design	14
				DES.ev	Design evaluation	3
FND	**Mathematical and engineering fundamentals**	**80**		**VAV**	**Software verification and validation**	**37**
FND.mf	Mathematical foundations	50		VAV.fnd	V&V terminology and foundations	5
FND.ef	Engineering foundations for software	22		VAV.rev	Reviews and static analysis	9
FND.ec	Engineering economics for software	8		VAV.tst	Testing	18
				VAV.par	Problem analysis and reporting	5
PRF	**Professional practice**	**29**		**PRO**	**Software process**	**33**
PRF.psy	Group dynamics and psychology	8		PRO.con	Process concepts	3
PRF.com	Communications skills (specific to SE)	15		PRO.imp	Process implementation	8
PRF.pr	Professionalism	6		PRO.pp	Project planning and tracking	8

				PRO.cm	Software configuration management	6
				PRO.evo	Evolution processes and activities	8
MAA	**Software modeling and analysis**	**28**		**QUA**	**Software quality**	**10**
MAA.md	Modeling foundations	8		QUA.cc	Software quality concepts and culture	2
MAA.tm	Types of models	12		QUA.pca	Process assurance	4
MAA.af	Analysis fundamentals	8		QUA.pda	Product assurance	4
REQ	**Requirements analysis and specification**	**30**		**SEC**	**Security**	**20**
REQ.rfd	Requirements fundamentals	6		SEC.sfd	Security fundamentals	4
REQ.er	Eliciting requirements	10		SEC.net	Computer and network security	8
REQ.rsd	Requirements specification and documentation	10		SEC.dev	Developing secure software	8
REQ.rv	Requirements validation	4				

Table 5 The ACM/IEEE Computer Engineering Body of Knowledge

CE-ALG Algorithms and Complexity [30 core hours] CE-ALG0 History and overview [1] CE-ALG1 Basic algorithmic analysis [4] CE-ALG2 Algorithmic strategies [8] CE-ALG3 Computing algorithms [12] CE-ALG4 Distributed algorithms [3] CE-ALG5 Algorithmic complexity [2] CE-ALG6 Basic computability theory	**CE-CAO Computer Architecture and Organization [63 core hours]** CE-CAO0 History and overview [1] CE-CAO1 Fundamentals of computer architecture [10] CE-CAO2 Computer arithmetic [3] CE-CAO3 Memory system organization and architecture [8] CE-CAO4 Interfacing and communication [10] CE-CAO5 Device subsystems [5] CE-CAO6 Processor systems design [10] CE-CAO7 Organization of the CPU [10] CE-CAO8 Performance [3] CE-CAO9 Distributed system models [3] CE-CAO10 Performance enhancements CE-CAO11 Crosscutting Issues
CE-CSE Computer Systems Engineering [18 core hours] CE-CSE0 History and overview [1] CE-CSE1 Life cycle [2] CE-CSE2 Requirements analysis and elicitation [2] CE-CSE3 Specification [2] CE-CSE4 Architectural design [3] CE-CSE5 Testing [2] CE-CSE6 Maintenance [2] CE-CSE7 Project management [2] CE-CSE8 Concurrent (hardware/software) design [2] CE-CSE9 Implementation CE-CSE10 Specialist systems CE-CSE11 System-level test and diagnosis CE-CSE12 Reliability and fault tolerance CE-CSE13 Error detecting and correcting codes	**CE-CSG Circuits and Signals [43 core hours]** CE-CSY0 History and overview [1] CE-CSY1 Electrical Quantities [3] CE-CSY2 Resistive Circuits and Networks [9] CE-CSY3 Reactive Circuits and Networks [12] CE-CSY4 Frequency Response [9] CE-CSY5 Sinusoïdal Analysis [6] CE-CSY6 Convolution [3] CE-CSY7 Fourier Analysis CE-CSY8 Filters CE-CSY9 Laplace Transforms

CE-DBS Database Systems [5 core hours] CE-DBS0 History and overview [1] CE-DBS1 Database systems [2] CE-DBS2 Data modeling [2] CE-DBS3 Relational databases CE-DBS4 Database query languages CE-DBS5 Relational database design CE-DBS6 Transaction processing CE-DBS7 Distributed databases CE-DBS8 Physical database design	**CE-DIG Digital Logic [57 core hours]** CE-DIG0 History and overview [1] CE-DIG1 Switching theory [6] CE-DIG2 Combinational logic circuits [4] CE-DIG3 Modular design of combinational circuits [6] CE-DIG4 Memory elements [3] CE-DIG5 Sequential logic circuits [10] CE-DIG6 Digital systems design [12] CE-DIG7 Modeling and simulation [5] CE-DIG8 Formal verification [5] CE-DIG9 Fault models and testing [5] CE-DIG10 Design for testability
CE-DSP Digital Signal Processing [17 core hours] CE-DSP0 History and overview [1] CE-DSP1 Theories and concepts [3] CE-DSP2 Digital spectra analysis [1] CE-DSP3 The discrete Fourier transform [7] CE-DSP4 Sampling [2] CE-DSP5 Transforms [2] CE-DSP6 Digital filters [1] CE-DSP7 Discrete time signals CE-DSP8 Window functions CE-DSP9 Convolution CE-DSP10 Speech processing	**CE-ELE Electronics [40 core hours]** CE-ELE0 History and overview [1] CE-ELE1 Electronic properties of materials [3] CE-ELE2 Diodes and diode circuits [5] CE-ELE3 MOS transistors and biasing [3] CE-ELE4 MOS logic families [7] CE-ELE5 Bipolar transistors and logic families [4] CE-ELE6 Design parameters and issues [4] CE-ELE7 Storage elements [3] CE-ELE8 Interfacing logic families and standard buses [3] CE-ELE9 Operational amplifiers [4] CE-ELE10 Circuit modeling and simulation [3] CE-ELE11 Data conversion circuits CE-ELE12 Electronic voltage and current sources CE-ELE13 Transistor amplifier design CE-ELE14 Power circuits CE-ELE15 Feedback in electronics CE-ELE16 Active filters CE-ELE17 Integrated circuit building blocks
CE-ESY Embedded Systems [20 core hours] CE-ESY0 History and overview [1] CE-ESY1 Embedded microcontrollers [6] CE-ESY2 Embedded programs [3] CE-ESY3 Real-time operating systems [3] CE-ESY4 Low-power computing [2] CE-ESY5 Reliable system design [2] CE-ESY6 Design methodologies [3] CE-ESY7 Tool support CE-ESY8 Embedded multiprocessors CE-ESY9 Networked embedded systems CE-ESY10 Interfacing and mixed-signal systems	**CE-HCI Human-Computer Interaction [8 core hours]** CE-HCI0 History and overview [1] CE-HCI1 Foundations of human-computer interaction [2] CE-HCI2 Graphical user interface [2] CE-HCI3 I/O technologies [1] CE-HCI4 Intelligent systems [2] CE-HCI5 Human-centered software evaluation CE-HCI6 Human-centered software development CE-HCI7 Interactive graphical user-interface design CE-HCI8 Graphical user-interface programming CE-HCI9 Graphics and visualization CE-HCI10 Multimedia systems
CE-NWK Computer Networks [21 core hours] CE-NWK0 History and overview [1] CE-NWK1 Communications network architecture [3] CE-NWK2 Communications network protocols [4] CE-NWK3 Local, wide area, and wireless networks [4] CE-NWK4 Client-server computing [3] CE-NWK5 Data security and integrity [4] CE-NWK6 Wireless and mobile computing [2] CE-NWK7 Performance evaluation CE-NWK8 Data communications CE-NWK9 Network management CE-NWK10 Compression and decompression	**CE-OPS Operating Systems [20 core hours]** CE-OPS0 History and overview [1] CE-OPS1 Design principles [5] CE-OPS2 Concurrency [6] CE-OPS3 Scheduling and dispatch [3] CE-OPS4 Memory management [5] CE-OPS5 Device management CE-OPS6 Security and protection CE-OPS7 File systems CE-OPS8 System performance evaluation

CE-PRF Programming Fundamentals [39 core hours] CE-PRF0 History and overview [1] CE-PRF1 Programming Paradigms [5] CE-PRF2 Programming constructs [7] CE-PRF3 Algorithms and problem-solving [8] CE-PRF4 Data structures [13] CE-PRF5 Recursion [5] CE-PRF6 Object-oriented programming CE-PRF7 Event-driven and concurrent programming CE-PRF8 Using APIs	**CE-SPR Social and Professional Issues [16 core hours]** CE-SPR0 History and overview [1] CE-SPR1 Public policy [2] CE-SPR2 Methods and tools of analysis [2] CE-SPR3 Professional and ethical responsibilities [2] CE-SPR4 Risks and liabilities [2] CE-SPR5 Intellectual property [2] CE-SPR6 Privacy and civil liberties [2] CE-SPR7 Computer crime [1] CE-SPR8 Economic issues in computing [2] CE-SPR9 Philosophical frameworks
CE-SWE Software Engineering [13 core hours] CE-SWE0 History and overview [1] CE-SWE1 Software processes [2] CE-SWE2 Software requirements and specifications [2] CE-SWE3 Software design [2] CE-SWE4 Software testing and validation [2] CE-SWE5 Software evolution [2] CE-SWE6 Software tools and environments [2] CE-SWE7 Language translation CE-SWE8 Software project management CE-SWE9 Software approaches and software fault tolerance	**CE-VLS VLSI Design and Fabrication [10 core hours]** CE-VLS0 History and overview [1] CE-VLS1 MOS Transistor Fundamentals [3] CE-VLS2 Processing and Layout CE-VLS3 Function of the Basic Inverter Structure [3] CE-VLS4 Circuit Characterization and Performance CE-VLS5 Combinational Logic Circuits CE-VLS6 Sequential Logic Circuits CE-VLS7 Alternative Circuit Structures/Low Power Design CE-VLS8 Semiconductor Memories and Array Structures [3] CE-VLS9 Chip Input/Output Circuits CE-VLS10 Semi custom Design Technologies CE-VLS11 ASIC Design Methodology

REFERENCES

Ardis, M. A., Budgen, D., Hislop, G. W., Offutt, J., Sebern, M. J., & Visser, W. (2015). SE 2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. *IEEE Computer*, *48*(11), 106-109.

Force, A. J. T. (2013). Computer science curricula 2013: Curriculum guidelines for undergraduate degree programs in computer science. *IEEE Computer Society*.

Lyon, D. A. (2004). Computer Engineering 2004–Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering: A Report in the Computing Curricula Series.

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker Jr, J. F., Sipior, J. C., & de Vreede, G. J. (2010). IS 2010: Curriculum guidelines for undergraduate degree programs in information systems. *Communications of the Association for Information Systems*, *26*(1), 18.