Incorporating CC2020 and SWECOM Competencies into Software Engineering Curricula: A Tutorial

John Impagliazzo

Pierre Bourque

Nancy R. Mead

ASEE&T

32nd CSEE&T

Munich, Germany

2020 November 10

John Impagliazzo

School of Engineering and Applied Science

Hofstra University

Hempstead, New York USA

john.impagliazzo@hofstra.edu

Pierre Bourque

École de Technologie Supérieure

Université du Québec

Montréal, Québec Canada

pierre.bourque@etsmtl.ca

Nancy R. Mead

Software Engineering Institute

Carnegie Mellon University

Pittsburgh, Pennsylvania USA

nrmcmu@gmail.com

Tutorial Timeline (Time: Activity [Person])

000: Tutorial introduction [J]

005: CC2020 project overview and competency [J]

020: SWECOM model overview and skills [P]

030: Industry perspectives, graduate attributes [P]

040: SWE CC2020 draft competencies [N]

050: Q&A on presentations [All]

055: Activity 1 instructions [J,P,N]

060: Breakout1: CC2020/SWECOM samples [A]

075: Discussion: Competencies for SWE [All]

090: PAUSE/BREAK

100: Activity2 instructions [N, P, J]

105: Breakout2: Create SWE competencies [A]

135: Report back [A, N]

155: Future SWE educational activities [All]

170: Closing remarks [P, N, J]

180: Adjournment [All]

A=Audience, N=Nancy, P=Pierre, J=John, All=Everyone

The CC2020 Project and Competency

[John]

Some Details on CC2020

Principal Sponsors: ACM … with IEEE Computer Society

Other Sponsors/Endorsers: ACM China, ACM SIGCSE China, ACM Europe, ACM India, AIS, AITP EDSIG, Italian Association of Computer Scientists, Informatics Europe, Informatics4All, Information Processing Society of Japan, ACM SIGCHI, Latin American Center on Computing (CLEI)

Large Investment: Approximately $200,000 USD

Task Force: 50 professionals, 20 countries, 6 continents

Objective: A modern replacement for CC2005

Contents of CC2020 Report (Draft at cc2020.net)

1. Introducing CC2020
2. Evolution of Computing Education
3. Knowledge-based Computing Education
4. Competency-based Computing Education
5. Visualizations
6. Global and Professional Considerations
7. Curricular Design – Challenges and Opportunities
8. Beyond CC2020

Appendix. Poster sample, Skills framework, Draft competency examples, Visualizations, Other

John – Principal Coauthor

CC2005

Best Known For

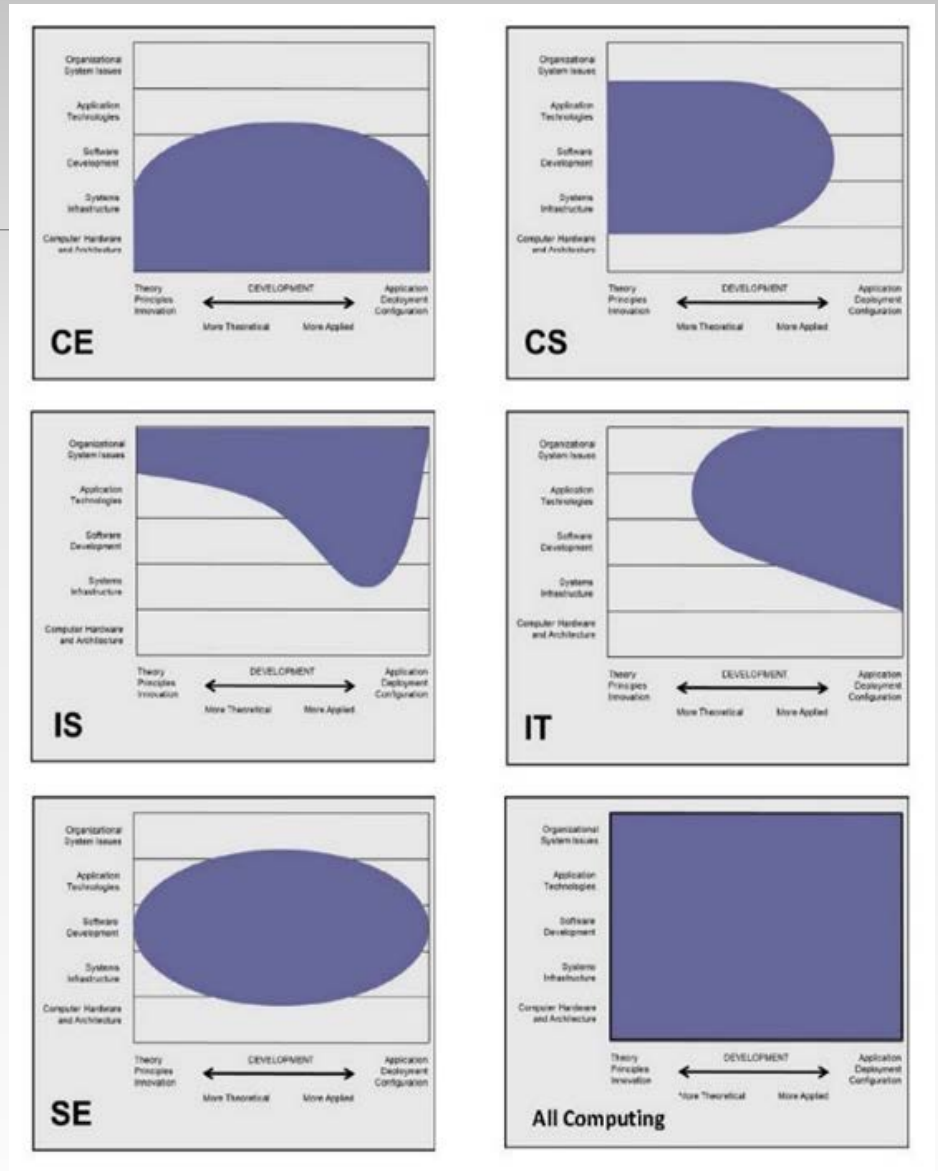
Comparative images and diagrams: →→→→

Comparison tables for five disciplines areas

Reflected Knowledge-based established curricula

For more information, go to:

www.acm.org → Education
→ Curricula Recommendations



Knowledge-based Learning

Content Knowledge

- Body of knowledge
- Information (facts, concepts, theories, and principles)
- Transfer to students
- Not necessarily concerned with related skills

Knowledge Base

- Collection of information about a subject

Knowledge-based Learning

- Methods of learning that teachers employ
- Optimal learning environments

Past and Current Situation

Body of Knowledge

Complete set of concepts, terms and activities

Make up a professional domain

It is more than simply a collection of terms … or a collection of information.

The accepted ontology for a specific domain.

[Institute for Competitive Intelligence]

Learning Outcomes

Statements of what a learner is expected *to know* and *demonstrate*

Occurs at the end of a learning unit

Units can be a course module, entire course, or full program

[IT2017 Report]

“Ka-Ku-Lo” Syndrome

Traditionally, curricula based on:

- Knowledge Areas (KAs)
- Knowledge Units (KUs)
- Learning Outcomes (LOs)

Shortfall:

- Focus only/mostly on knowledge
- Achievement measured by test results
- Student lacks maturity in performance
- Not beneficial for today’s computing world

Technology (Digital) Skills Gap

80% of Americans (U.S.) agree there is a skills gap, and 35% say it affects them personally.

42.5% of recent graduates were underemployed as of March 2018, according to the Federal Reserve Bank of New York.

$160 billion. This is the total cost of the skills gap to US companies per year.

60% of U.S. employers have job openings that stay vacant for 12 weeks or longer. The average cost from job vacancies is $800,000 or more annually.

81% of employers said prospective employees lack critical thinking and analytical reasoning skills. Also, …

75% think graduates lack adequate innovation and diversity skills.

PSI Select International
https://blog.psionline.com/5-stats-about-the-skills-gap-that-demands-attention

STEM Job Comparison

Workforce USA	2018 Jobs	2028 Jobs	Job Increase	Percent Increase
All Jobs	161,037,700	169,435,900	8,398,100	5.2
STEM Jobs	9,708,300	10,566,800	858,500	8.8
Non-STEM Jobs	151,329,400	158,869,100	7,539,600	5.0

Bureau of Labor Statistics, 2018 Sep 04

Situation!

Which percent of current graduates from undergraduate computing or engineering programs continue full-time their formal education?

50%? 20%? 10%? 5%? 1%?

The remaining graduates …

50% 80% 90% 95% 99%

… go to industry or government employment

Competency

Information Technology

Competence refers to the *performance* standards associated with a profession or membership to a licensing organization.

[IT2017 Report

Software Engineering

[Competency is the] demonstrated ability *to perform work activities* at a stated competency level, which is one of five increasing levels of ability to perform an activity …

[SWE Competency Model – 2014]

Competency Meaning [Industry Perspective]

Competency = Dispositions

+ Skills

+ Knowledge

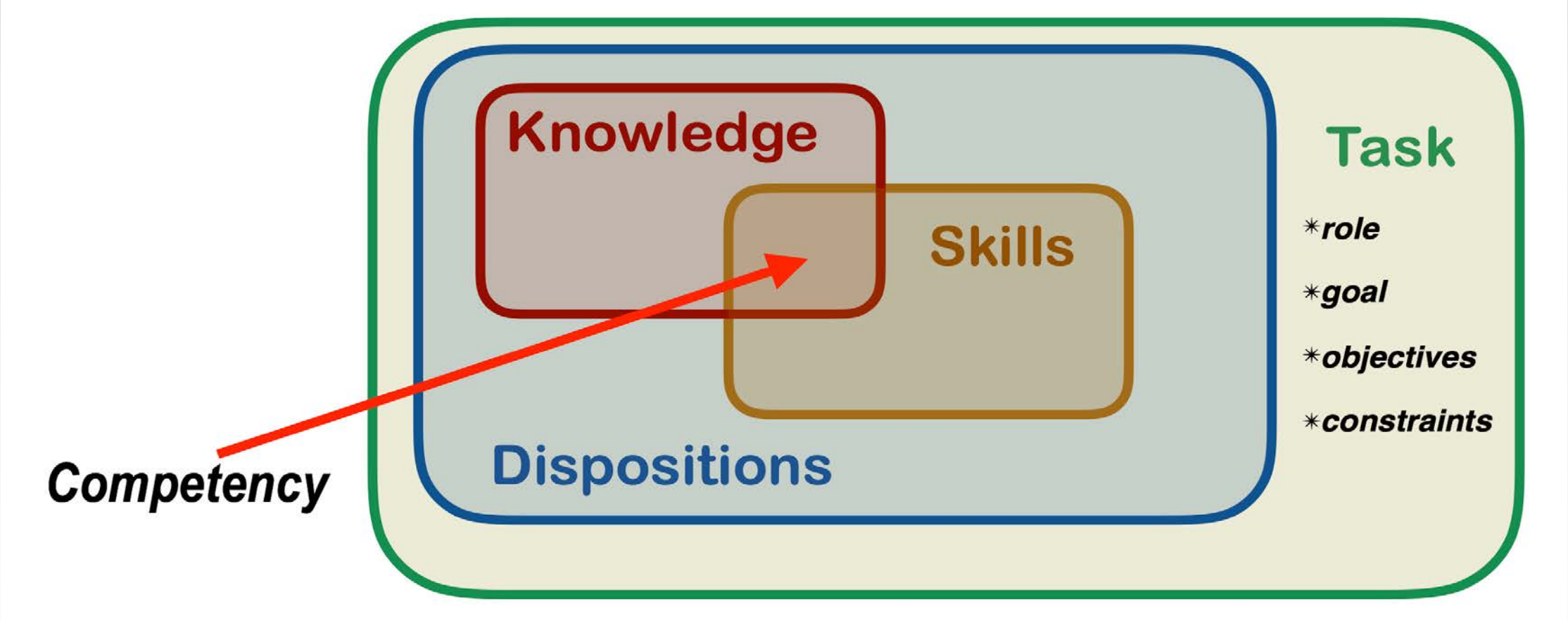
in Context / Task

Competency = Human Attributes + Technical Skills + Knowledge

Competency = Behaviors + Technical Skills + Knowledge

Competency = Abilities + Technical Skills + Knowledge

Competency in Task (From CC2020 Draft Report)



Six Computing Knowledge Categories
(From CC2020 Draft Report)

Thirty-four computing knowledge elements partitioned into six knowledge categories

1. Humans and Organizations
2. Systems Modeling
3. Software Systems Architecture
4. Software Development
5. Software Fundamentals
6. Hardware

Thirteen Professional Knowledge Elements

(From CC2020 Draft Report)

1. Analytical and Critical Thinking
2. Collaboration and Teamwork
3. Ethical and Intercultural Perspectives
4. Mathematics and Statistics
5. Multi-Task Prioritization and Management
6. Oral Communication and Presentation
7. Problem Solving and Trouble Shooting
8. Project and Task Organization and Planning
9. Quality Assurance / Control
10. Relationship Management
11. Research and Self-Starter/Learner
12. Time Management
13. Written Communication

Six Cognitive Skill Levels
(From CC2020 Draft Report)

1. Remembering
2. Understanding
3. Applying
4. Analyzing
5. Evaluating
6. Creating

Eleven Dispositional Elements
(From CC2020 Draft Report)

1. Proactive
2. Self-directed
3. Passionate
4. Purpose-driven
5. Professional
6. Responsible
7. Adaptable
8. Collaborative
9. Responsive
10. Meticulous
11. Inventive

Competency Cluster Example

Competency Statement Identify and document system requirements by applying a known requirements elicitation technique in work sessions with stakeholders, using facilitative skills, as a contributing member of a requirements team.	
Knowledge Element	**Skill Level**
Requirements Analysis	Evaluating
Oral Communication	Applying
Written Communication	Applying
Teamwork and Collaboration	Applying
Disposition(s)	
Purpose-driven	Responsible Collaborative

From 2018 Publication in China

专栏 中国计算机学会通讯 第 14 卷 第 9 期 2018 年 9 月

能力的作用与计算机教育的未来

John Impagliazzo
霍夫斯特拉大学

关键词：计算机教育　能力

2018 年 7 月 20 日，中国计算机学会 (CCF) 在南京举办了未来计算机教育峰会 (the Future Computer Education Summit, FCES)，我应邀在会上作主题演讲。本文是该演讲的一个延伸。该演讲与峰会的主题息息相关。在技术导向型社会，未来的计算机教育一定是基于能力的，单纯依靠获取知识的方

业内求职。然而，计算机行业的雇主们对雇员的需求和表现给予很高的期望，他们几乎都希望雇员就职后可以立即发挥才能，帮助企业盈利。因为雇主不考虑毕业生的学术成就和知识导向成就，而且毕业生缺乏行业所期望和要求的专业技能和人格，所以他们往往难以找到有意义的工作。学生在校期间

The Role of Competencies and the Future of Computing Education

专栏 中国计算机学会通讯 第14卷 第9期 2018年9月

能力的作用与计算机教育的未来

John Impagliazzo
霍夫斯特拉大学

关键词：计算机教育　能力

2018年7月20日，中国计算机学会(CCF)在南京举办了未来计算机教育峰会(the Future Computer Education Summit, FCES)，我应邀在会上作主题演讲。本文是该演讲的一个延伸。该演讲与峰会的主题息息相关。在技术导向型社会，未来的计算机教育一定是基于能力的，单纯依靠获取知识的方

业内求职。然而，计算机行业的雇主们对雇员的需求和表现给予很高的期望，他们几乎都希望雇员就职后可以立即发挥才能，帮助企业盈利。因为雇主不考虑毕业生的学术成就和知识导向成就，而且毕业生缺乏行业所期望和要求的专业技能和人格，所以他们往往难以找到有意义的工作。学生在校期间

SWECOM
Overview and Skills

[Pierre]

SWECOM
Software Engineering Competency Model

Software Engineering Competency Model (SWECOM), 2014. IEEE Computer Society. ISBN-13: 978-0-7695-5373-3 *

Based on consensus-based standards and reference documents

Can be tailored to your needs

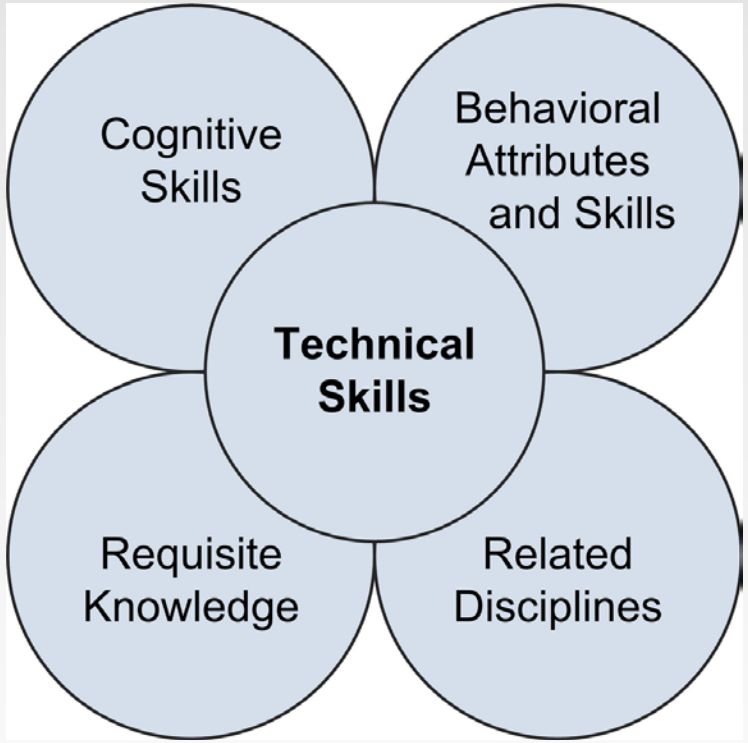
Does not prescribe or recommend any tools, process approaches, development methods and so forth

Covers technical skills but does not include project management or general management skills

One of the intended audiences of SWECOM is curriculum designers

Available at https://www.computer.org/volunteering/boards-and-committees/professional-educational-activities/software-engineering-competency-model

Elements of SWECOM



Two Types of Technical Skill Areas

Life cycle Skill Areas

- Includes skills needed to accomplish various work activities within a phase of software development or sustainment.

Crosscutting Skill Areas

- Applies across all life cycle skill areas and, in some cases, a crosscutting skill may apply to other crosscutting skill areas.
- Crosscutting skill areas are sometimes called “specialty disciplines” that are practiced by specialists in those skill areas.

Skill Areas

Life Cycle Skill Areas	Software Crosscutting Skills Areas
Software Requirements Skills	Software Process and Life Cycle Skills
Software Design Skills	Software Systems Engineering Skills
Software Construction Skills	Software Quality Skills
Software Testing Skills	Software Security Skills
Software Sustainment Skills	Software Safety Skills
	Software Configuration Skills
	Software Measurement Skills
	Human-Computer Interaction Skills

Skills by Skill Area

Software Requirements Skills Area	Skills
References: [ACM 2004] [Laplante 2009] [Robertson 2012] [SWEBOK 2014] [Wiegers 2013]	Software Requirements Elicitation Software Requirements Analysis Software Requirements Specification Software Requirements Verification and Validation Software Requirements Process and Product Management

Competency Levels

Activities are specified at five levels of competency:

- Technician
- Entry Level Practitioner
- Practitioner
- Technical Leader
- Senior Software Engineer

Following notations are also used

- Follows (F)
- Assists (A)
- Participates (P)
- Leads (L)
- Creates (C)

Software Requirements Skill Sets and Activities by Competency Level					
Skill Area: Software Requirements					
Skill: Requirements Specification					
Competency Levels	Technician	Entry Level Practitioner	Practitioner	Team Leader	Senior Software Engineer
Activities	1. Assists with preparation of requirements for consistency with internal and published standards. (F/A)	1. Prepares requirements documentation including descriptions of interfaces and functional and non- functional requirements. (P)	1. Selects the most appropriate formal and informal notations for describing interfaces and functional and non-functional requirements. (P/L)	1. Leads development of the SRS. (L)	1. Creates new requirements specification methods. (C)

Subset of the Table

Industry Perspectives and Graduate Attributes

[Pierre]

Industry Competency Models

Skills Framework for the Information Age (SFIA)

European e-Competence Framework (e-CF)

i Competency Dictionary (iCD)

US Department of Labor Information Technology (US IT) Competency Model

Graduate Attributes

A key component of the Washington Accord

Incorporated into the Accreditation Criteria of the Canadian Engineering Accreditation Board (CEAB)*

- CEAB accredits undergraduate engineering programs across Canada and enables graduates of these programs “to begin the process to be licensed as professional engineers.”

* https://engineerscanada.ca/accreditation/accreditation-board

Graduate Attributes

1- A knowledge base for engineering	7- Communication skills
2- Problem analysis	8- Professionalism
3- Investigation	9- Impact of engineering on society and the environment
4- Design	10- Ethics and equity
5- Use of engineering tools	11- Economics and project management
6- Individual and team work	12 Life-long learning

Graduate Attribute Indicators – Course on Software Requirements at ÉTS

Indicators: "For each attribute, there must be a set of measurable, documented indicators that describe what students must achieve in order to be considered competent in the corresponding attribute."

- GA2 - Indicator i1: Identify and formulate complex problems by establishing the context, parameters and constraints.
- GA3 - Indicator i1: Plan the investigation process using literature reviews, existing solutions, prototypes, experiments, simulations, etc.
- GA7 - Indicator i1: Professionally write technical reports effectively documenting engineering work using different forms of language (natural, graphic, mathematical, etc.).

These indicators are established at the institution-level for all engineering programs. They can be replaced, and others can be added as well at the program-level.

- **An agreed upon set of competencies for a software engineering undergraduate program would be most helpful in such an exercise and in interpreting these indicators at the course-level when selecting assessment tools and for analyzing results.**

Software Engineering
CC2020 Draft Competencies

[Nancy]

Software Engineering [1 of 2]

As we know (From CC2020 Draft Report) :

- Software Engineering is an engineering discipline that emphasizes development and use of rigorous methods to design and construct software artifacts that will reliably perform specified tasks
- People with a “software engineer” job title may or may not have a degree in software engineering
- Software Engineering coursework and specialties may be embedded in other degree programs such as computer science or computer engineering

Software Engineering [2 of 2]

In addition:

- The software engineering competencies in CC2020 drew on many sources, such as SE2014 and SWECOM as well as Master of Software Engineering and Software Assurance curricula
- The SWE competencies also reflected prior curriculum development experience and first-hand knowledge of the subgroup team members

SE Knowledge Domains
(From CC2020 Draft Report)

CC2020 SE Subgroup Created 56 Draft Competencies from the following 15 Knowledge Domains

- Behavioral Attributes
- Human-Computer Interaction
- Project Management
- Software Configuration Management
- Software Construction
- Software Design
- Software Measurement
- Software Process and Life Cycle
- Software Quality
- Software Requirements
- Software Safety
- Software Security
- Software Sustainment
- Software Systems Engineering
- Software Testing

Software Requirements Draft Competencies

(From CC2020 Draft Report) [1 of 2]

1. Identify and document software requirements by applying a known requirements elicitation technique in work sessions with stakeholders, using facilitative skills, as a contributing member of a requirements team.
2. Analyze software requirements for consistency, completeness, and feasibility, and recommend improved requirements documentation, as a contributing member of a requirements team.
3. Specify software requirements using standard specification formats and languages that have been selected for the project, and be able to describe the requirements in an understandable way to non-experts such as end users, other stakeholders, or administrative managers, as a contributing member of a requirements team.

Software Requirements Draft Competencies

(From CC2020 Draft Report) [2 of 2]

4. Verify and validate the requirements using standard techniques, including inspection, modeling, prototyping, and test case development, as a contributing member of a requirements team.
5. Follow process and product management procedures that have been identified for the project, as a contributing member of the requirements engineering team.

Software Security Draft Competencies

(From CC2020 Draft Report) [1 of 2]

1. Apply the project's selected security lifecycle model (e.g. Microsoft SDL), as a contributing member of a project team.
2. Identify security requirements by applying the selected security requirements method, as a contributing member of a software project team.
3. Incorporate security requirements into architecture, high-level, and detailed design, as a contributing member of a software project team.
4. Develop software using secure coding standards.

Software Security Draft Competencies

(From CC2020 Draft Report) [2 of 2]

5. Execute test cases that are specific to security.
6. Adhere to the project's software development process, as a contributing member of a software project team.
7. Develop software that supports the project's quality goals and adheres to quality requirements.

Project Management Draft Competencies
(From CC2020 Draft Report)

1. Explain the principal elements of management for a small project team.
2. Assist in the managerial aspects of a small project team, including software estimation, project planning and tracking, staffing and resource allocation, and risk management.
3. Develop and implement plans for measurement of software processes and work products using appropriate methods and tools.
4. Work effectively with other team members in project management activities.

Q & A
on
Presentations

[All]

Activity 1 Instructions

[John, Pierre, Nancy]

Instructions for Activity 1

- Split into teams as directed
- Read ASEET - Background-Instructions
- Read Sample SWE competencies: Example 1 and Example 2, pp. 9-10
- Use the next slide to reflect on the sample competencies
- Identify a team leader who will report the result of the team's discussions

Consider the Following Points

Read the two example competencies and consider the following:

1. When you read each example competency, do you nod your head in agreement or scratch your head trying to figure out the point?
2. Are the competencies written at a consistent level with one another?
3. How well do they relate to the SWECOM material presented earlier?

BREAKOUT 1
CC2020/SWECOM Samples

[Audience]

DISCUSSION
Competencies for SWE

[Audience]

PAUSE/BREAK

[All]

Activity 2 Instructions

[Nancy, Pierre, John]

Instructions for Activity 2

- Go to your assigned breakout room to form teams
- Use ASEET-Background-Instructions pages 12-14 to select specific competencies to work on within your assigned competency areas
- Use ASEET-Participant-Worksheet
- Identify a team leader and fill out worksheet(s) for each selected competency using reference materials
- Send completed worksheet(s) to marie.lefebvre@etsmtl.ca

Instructions for Activity 2

Breakout rooms will work on competencies within the following competency areas:

- Software Design + Systems engineering
- Software Construction + Measurement
- Software Quality + Safety
- Software Configuration + Security
- Software Sustainment + Testing
- Software Process + Requirements

BREAKOUT 2
Create SWE Competencies

[Audience]

REPORT BACK

[All]

Future Software Engineering Educational Activities

[All]

CLOSING REMARKS

[Pierre, Nancy, John]

Thank You !

ADJOURNMENT

[All]