



NITIC

National Information Technology
Innovation Center

Business Industry Leadership Team (BILT)

Software Development KSAs June 2025

This prioritized Knowledge, Skills, and Abilities (KSA) list was developed by the National IT Innovation Center (NITIC) in collaboration with IT industry subject matter experts (SMEs). This list is intended to help faculty and administrators align curriculum with industry needs so graduates are “workforce ready” when they graduate.

SMEs convened online to rank each KSA item one by one – a ranking of “4” meant that item was essential for entry-level IT workers, while a “1” meant that item could be removed from program curriculum. By default, items with an average vote of 2.6 or lower were turned pink to signal that this item may not be worth keeping.

After the vote, the SMEs discussed the results as a group. This discussion led to some revisions in the KSA descriptions and rankings. Some “pink” items the SMEs together deemed important despite the lower average and were kept, while some “green” items with a higher average the SMEs recommended be removed from the list. Those items can be found at the bottom of each KSA section under “Recommended Removal.”

The “To Be Included in the Next Vote” section at the bottom includes items the SMEs suggested that were not included in the original list vote.

This KSA prioritization process is a cornerstone of the successful Business and Industry Leadership Team (BILT) model which puts businesses in a co-leadership role.

Learn more about how the BILT works – and how you can implement it with your own program – by using these resources:

NITIC orientation video; www.tiny.cc/BILTorient
Convergence Technology Center BILT handbook: www.tiny.cc/implementingBILT

Knowledge		# votes (4 = most important)				
		4	3	2	1	Avg
K-1	Knowledge of software development models (e.g. Waterfall Model, Spiral Model, Agile).	5	7	2	0	3.21
K-2	Knowledge of system design tools, methods, and techniques, including automated systems analysis and design tools.	4	8	2	0	3.14
K-3	Knowledge of effective software debugging principles.	6	5	4	0	3.13
K-4	Knowledge of computer programming languages and principles in general.	11	3	1	0	3.67
K-5	Knowledge of web services (e.g. service-oriented architecture, REST, and web service description language).	6	6	2	0	3.29
K-6	Knowledge of visual representations of a program or system.	3	7	4	0	2.93
K-7	Knowledge of how programs communicate across networks using asynchronous and synchronous techniques. (when to use and why) (e.g., REST, GraphQL)	7	5	2	0	3.36
K-9	Knowledge of event handling in a GUI.	3	11	0	0	3.21
K-10	Knowledge of automated regression testing.	5	4	3	2	2.86
K-11	Knowledge of the appropriate use of cookies.	3	3	7	1	2.57
K-13	Knowledge of two or more operating systems that are current industry standards (e.g. Linux, Windows Apple OS).	6	2	4	2	2.86
K-14	Knowledge of error handling constructs.	5	8	1	0	3.29
K-15	Knowledge of the differences between client-side scripting and server-side scripting.	4	6	3	1	2.93
K-16	Knowledge of common program architectures (e.g. standalone, three-tier, web-based, cloud-based, serverless, microservice).	7	4	1	1	3.31
K-17	Knowledge of the local development cycle (e.g. build, deploy, test, debug).	7	4	2	1	3.21
K-18	Knowledge of server software patterns, messaging patterns both async and synch.	3	6	5	0	2.86
K-19	Knowledge of database integration/management software.	5	4	5	0	3.00
K-20	Knowledge of applying AI and ML methods and algorithms.	6	6	1	2	3.07
K-21	Knowledge of software collaboration tools (e.g. version control, bug tracking, continuous integration).	5	6	2	1	3.07

K-22	Knowledge of the limits vs actual process of continuous integration and production deployment practices of devsecops/devnetsecops.	1	5	6	2	2.36
K-23	Knowledge of cybersecurity and privacy principles and methods that apply to software development.	6	8	1	0	3.33
K-24	Knowledge of system and application security threats and vulnerabilities (e.g., buffer overflows, cross-site scripting, SQL injection, race conditions, insecure deserialization, SSRF, CSRF, supply chain attacks, API abuse, replay attacks, return-oriented attacks, and malicious code).	8	3	3	0	3.36
K-25	Knowledge of code security (e.g. hashing, encryption, cryptography, threat modeling).	9	2	4	0	3.33
K-26	Knowledge of Privacy Impact Assessments in terms of privacy and identity management.	3	6	4	1	2.79
K-27	Knowledge of cyber threats and vulnerabilities.	4	6	4	0	3.00
K-28	Knowledge of software related information technology (IT) security principles and methods (e.g. modularization, layering, abstraction, data hiding, simplicity/minimization).	4	6	4	0	3.00
K-29	Awareness of standards such as PCI, PHI, and GDPR.	2	8	2	2	2.71
K-30	Knowledge of basic security practices including threats and vulnerabilities that may arise from interactions with other systems, external and legacy code.	6	3	4	1	3.00
K-31	Knowledge of computer network fundamentals (e.g. TCP/IP, HTTPS, ports, firewall, LAN/WAN etc.)and network security methodologies.	2	9	3	0	2.93
K-32	Knowledge of implementation and utilization of cloud services including deployment (e.g. AWS, Microsoft Azure).	5	5	3	1	3.00
K-33	Awareness of cloud computing concepts (e.g. IoT, edge computing).	4	6	2	1	3.00
K-35	Knowledge of the difference between AI and ML.	5	6	2	1	3.07
K-36	Awareness of current and specialized AI and ML tools and their application to business problems.	3	7	3	1	2.86
K-37	Conceptual knowledge of PKI and PQA.	2	5	6	1	2.57
K-38	Knowledge of DevSecOps concepts and components.	3	4	5	2	2.57
K-39	Knowledge of structured and unstructured data sources.	4	6	4	0	3.00

K-40	Knowledge of open source software and risks involved.	4	7	1	2	2.93
K-41	Knowledge of ethics and its application to software development.	7	5	2	1	3.20
K-42	Knowledge of best practices for Design/UI/UX/accessibility as applied to software development.	4	5	5	0	2.93
K-43	Knowledge of lifecycle development/steady state/end of life.	6	3	1	4	2.79
K-44	Knowledge of mobile application development.	5	1	8	1	2.67
K-45	Knowledge of how to protect data privacy through code.	6	5	1	2	3.07
K-46	Knowledge of process flow and how the upgrade/implementation of software is accomplished through definitive understanding of team collaboration in DevOps, End of Life Cycle, and including importance of foundational security.	3	7	2	2	2.79
K-47	Knowledge of performing integrated quality assurance testing for security functionality and resiliency attack.	2	8	1	3	2.64
K-48	Knowledge of how to identify security implications in the software acceptance phase including completion criteria, risk acceptance and documentation, common criteria, and methods of independent testing and report concerns to IT/software team.	3	5	5	1	2.71
K-49	Knowledge of communication encryption, including PKI and PQA.	2	4	7	1	2.50
K-50	Knowledge of how to identify and leverage the enterprise-wide security services while designing and developing secure applications (e.g., Enterprise PKI, Federated Identity server, PQA) when appropriate.	1	6	5	2	2.43
K-51	Knowledge of how to identify and analyze user needs and use needs to establish a plan in the selection, creation, evaluation, implementation and administration of information technology systems.	4	5	3	1	2.92
K-52	Knowledge of security requirements into application design elements including documenting the elements of the software attack surfaces, conducting threat modeling, and defining any specific security criteria.	3	5	3	2	2.69

K-53	Knowledge of architecture patterns and when to use them to build applications.	6	5	2	0	3.31
K-54	Knowledge of algorithms and data structures (e.g. big-O, linked lists, hash maps, sorting, etc.).	7	6	0	0	3.54
K-55	Knowledge of Binary search tree and how binary search works	3	8	2	0	3.08
K-56	Knowledge of Agile development practices including the use of user stories to capture requirements and guide iterative development.	7	3	2	1	3.23
K-57	Knowledge of how AI-assisted development tools (e.g., code completion, documentation generation, test case synthesis) integrate into modern software workflows, including their benefits, limitations, and responsible use practices.	7	5	1	1	3.29
Recommended Removal - Skills						
K-8	Knowledge of Software Integration Management Systems – how industry documents final product builds to show all of the elements that have changes and checks those that have not changed.	1	3	6	0	2.50
K-12	Knowledge of how applets differ from applications in terms of program form, operating context, and how they are started.	0	6	5	3	2.21
K-34	Knowledge of software development and implementation for communicating and gathering data from IoT devices.	2	4	6	2	2.43
Skills						
S-1	Skill in using built-in functions as well as skill in creating custom functions, subroutines, and procedures within software using scripting languages.	7	6	0	0	3.54
S-2	Skill in integrating standard object model components with server pages in support of the User Experience.	3	8	2	0	3.08
S-3	Skill in conducting software debugging.	8	3	3	0	3.36
S-4	Skill in creating programs that validate and process multiple inputs including command line arguments, environmental variables, and input streams.	7	5	1	0	3.46
S-5	Skill in writing code in current programming languages and frameworks.	11	1	1	0	3.77

S-6	Skill in developing applications that can log and handle errors, exceptions, and application faults and logging.	5	5	3	0	3.15
S-7	Skill in applying root cause analysis (RCA) techniques to solving software/customer issues.	4	7	1	1	3.08
S-8	Skill in the live production environment (e.g. monitoring, logging, alerting, remote debugging).	3	7	2	1	2.92
S-9	Skill in using electronic mail software (e.g. Google Gmail; IBM Notes Hot technology; Microsoft Exchange Server Hot technology; Microsoft Outlook Hot technology).	7	1	2	3	2.92
S-10	Skill in using graphical user interface development software (e.g. Graphical user interface GUI builder software; Graphical user interface GUI design software).	2	5	6	0	2.69
S-11	Skill in applying object-oriented and component-based programming principles using modern languages and frameworks (e.g., Python, Java, C++, JavaScript with React or Angular), and in working with structured data formats such as JSON or XML.	8	4	1	0	3.54
S-12	Skill in creating classes that use inheritance aspects of the object-oriented paradigm.	6	4	3	0	3.23
S-13	Skill in using, incorporating and utilizing cookies.	0	7	3	3	2.31
S-14	Skill in implementing programs that use local or remote databases with standard protocols.	6	5	2	0	3.31
S-15	Skill in using a scripting language on the server side and the client side of a distributed program.	5	4	3	1	3.00
S-16	Skill in evaluating and reporting software needs, constraints, analysis for application-specific concerns.	1	9	3	1	2.71
S-17	Skill in implementing levels of security in distributed software applications and applets.	2	7	3	1	2.77
S-18	Skill in deploying secure software according to secure software deployment methodologies, tools, and practices (e.g. PCI, GDPR, HIPAA, CCPA).	4	5	4	1	2.86
S-19	Skill in mobile application development.	3	4	6	1	2.64
S-20	Skills such as time management, risk management.	6	4	2	1	3.15
S-21	Skill in incorporating user experience feedback into software.	5	3	2	3	2.77

S-22	Skill in integrating third party open source resources into software including minimizing risk.	3	6	3	1	2.85
S-23	Skill in learning new and/or industry standard tools involved in the development of software.	7	4	2	0	3.38
S-25	Skill in engaging with users and understanding their user experience.	6	3	5	0	3.07
S-26	Skill in applying test-first approaches such as Test-Driven Development (TDD) and Behavior-Driven Development (BDD) to improve code quality and ensure software behavior aligns with business expectations.	4	6	3	1	2.93
S-27	Skill in applying secure development principles to integrate security throughout the software development lifecycle (SDLC).	5	8	0	1	3.21
S-28	Skill in using AI tools to augment software development tasks (e.g., writing code, generating tests, debugging), while applying knowledge of the programming language and context to critically evaluate and validate generated output.	6	7	0	1	3.29
S-29	Skill in applying secure development principles when contributing to secure design reviews.	3	8	2	1	2.93
S-30	Skill in leveraging AI development tools to enhance productivity while ensuring the correctness, security, and maintainability of code through manual review and domain expertise.	8	5	0	1	3.43
Recommended Removal - Skills						
S-24	Skill in producing technical content for tech writers.	0	5	7	1	2.31
To Be Included in Next Vote						
	Knowledge of AI-assisted programming (aka vibe coding)					
	Knowledge of System integration (e.g. JSON, REST, SOAP)					
	Knowledge of Data management and manipulation					
	Knowledge of Identity management solutions (MFA and 2F)					
	Knowledge of Low code integration with AI					