National BILT Meeting Minutes
"Software Development" Job Skills Vote and Discussion

MEETING DATE: **Tuesday, June 3, 2025**	MEETING TIME: 11:30am-1:00pm Eastern	MEETING PLACE: Zoom
RECORDER: Mark Dempsey	RECORDING: Available upon request	PREVIOUS MEETING: Industry trends meeting – April 29, 2025

MEMBERS PRESENT

BILT:		
Scott Andersen	Otto Dobretsberger, CODE Group	Milestone Technologies
Ali Asghari	Darryl Dunnington, DSD Inc	Mark Richter, Hitachi Delivery Systems
Laurabeth Baker, CODE Group	Mark Meyer, Omiga Solutions	Harvendra Singh
Vishi Singh Bhatia	Amit Prasad	Scott Whitmire
EDUCATORS: Brazosport College, Centerville HS, College of Dupage, Collin College, Columbus State Community College, Daytona State College, Forsyth Tech, Houston Community College, Johnson Community College, Kilgore College, Lansing Community College, Mass Bay Community College, Northern Virginia Community College, Northwest Arkansas Community College.		
NITIC STAFF: Ann Beheler, Mark Dempsey, Christina Titus, Larry McWherter (NITIC principal investigator), Alie Hernandez, Stephanie Schuler, Deb Hecht, Diane Meza, Andie Bonskiwske		

Agenda items	Discussion
NITIC and BILT overview	Larry provided an overview of the NITIC grant, then explained that the goal of today's meeting is to prioritize job skills for entry-level software development technicians in the next 12-46 months. Larry welcomed the BILT members and told them their anonymized feedback will be widely disseminated to educators nationwide. He also noted educators on the Zoom call are invited to primarily listen actively. Ann next provided an overview of the BILT (Business and Industry Leadership Teams). BILTs are business advisory councils that put employers in a co-leadership role. BILTs have two goals – aligning curriculum to workforce needs and building and deepening the relationship between educators and employers. BILTs host two kinds of meetings – cross-disciplinary "future trends" meetings and single-discipline job skills vote meetings like the one we're having today. Ann noted that the discussion after today's vote is just as important as the vote itself. Ann explained the 1-4 voting scale. Each job skill item will receive its own individual vote. * "4" means absolutely must be in the curriculum * "3" means should be in curriculum * "2" means it's a "nice to have" * "1" essentially means delete Christina pointed out that "knowledge" items refer to an awareness. Knowing about something does not mean you know how to do it. "Skill" items refer to practical know-how. Using a QR code (and URL in the chatbox), the SMEs voted electronically.

	Anything scoring 2.6 or less will be discussed because, typically, items with scores that low are left out of the curriculum.
Knowledge items	***K8 - Knowledge of Software Integration Management Systems, 2.50 average*** One employer noted she would not expect an entry-level hire to know this. No one disagreed. **ACTION: consider removing K8**. ***K11 - Knowledge of the appropriate use of cookies, 2.57 average*** One employer noted that an entry-level person would not need to know much about cookies, but another employer countered that appropriate use of cookies is important in web development. Another agreed that a better understanding of cookie creation and usability is important to avoid creating vulnerabilities. But he did acknowledge it depends on the job – web application developers will need to know this, but other development jobs would not. ***K12 - Knowledge of how applets differ from applications in terms of program form, operating context, and how they are started, 2.21 average*** One employer stated that applets are not up-to-date technology in the Java sphere. **ACTION: consider removing K12**. ***K20 - Knowledge of AI and ML methods and algorithms, 3.07 average*** One employer didn't think entry-level workers need to know this. New hires will not implement high-level algorithms. One employer agreed, but another argued that students need to know the basics. Maybe they don't know the algorithms, but they should know when to use one versus the other. He also noted he's seen a lot of inappropriate use of machine learning. For what kinds of problems do you use AI versus ML? Understanding concepts and methods is what's important here; students need to know how to use AI and ML, not build the tools. Another employer suggested that 36 months from now, half of the graduates seeking development jobs are not going to get one – they'll have to know AI and ML to get past the first interview. Junior developers will be using AI. AI will allow developers to analyze data much faster. He suggested that K20 also include "knowledge of applying" AI and ML. **ACTION: consider rewording K20 to "Knowledge of applying AI and ML methods and algorithms."** ***K22 - Knowledge of the limits vs actual process of continuous integration and production deployment practices of DevSecOps/DevNetSecOps, 2.36 average*** One employer suggested knowing theoretical limits isn't as important as encountering actual limits when you're coding, whether in school or on a job. You learn the limits when you do the work. Another employer noted that knowing how "continuous integration" works in the software development process, not just in relation to limits, is a knowledge area that will make someone more appealing as a candidate. ***K26 – Knowledge of Privacy Impact Assessments in terms of privacy and identity management, average 2.79*** One employer emphasized that students need to understand the impact of privacy and laws like the GDPR. Other BILT members agreed. ***K34 - Knowledge of software development and implementation for communicating and gathering data from IoT devices, 2.43 average***

	One employer wasn't sure IoT concepts need to be taught in a two-year program. Another noted that there are two pieces in K34 – the software "development and implementation" knowledge, and also the "communicating and gathering" knowledge. He argued that entry-level workers will probably learn how to gather IoT data on the job. He ranked this one low. **ACTION: consider removing K34.** ***K37 - Conceptual knowledge of PKI, 2.57 average*** One employer noted that authenticating and securing endpoints are too important to leave out of curriculum. Others agreed that basic knowledge of PKI is important, especially for enterprise solutions. A deeper understanding of implementation is probably not as essential. Another employer suggested this is like the earlier AI-ML discussion – students may not need to know about implementation, but they need to know what it is, why it works, and how to use it. The BILT then discussed the need to add PQA (post-quantum algorithms) and zero-trust to K37. PKI won't be around much longer. PQA will be everything. He believed that adding PQA to this item would make the vote average go way up. No one disagreed, although one employer pointed out many systems are still built around PKI. Knowing PKI is still helpful. Students should know both. From the chatbox: "PKI is almost legacy. Three years out, PQA will be necessary." PKI versus PQA: https://arqit.uk/hubfs/Facing_the_quantum_threat_Final_June_2022.pdf **ACTION: consider rewording K37 to "Conceptual knowledge of PKI and PQA."** ***K38 - Knowledge of DevSecOps, 2.57 average*** The BILT wasn't sure how deep the knowledge on this should be. One employer suggested students need to know what DevSecOps does – integrates and automates the build process, and analyzes the code before it gives you a final product. He proposed that the most important thing for students to understand is how DevSecOps provides code feedback. Another employer suggested providing context: why do students need to know this and how is it important to their career? Another employer noted that DevSecOps didn't exist 20 years ago and will likely have a much smaller role soon because something else will come along to replace it. Students should understand that reality. The BILT all liked one suggestion from another employer: "Knowledge of DevSecOps concepts and components." **ACTION: consider rewording K38 to "Knowledge of DevSecOps concepts and components."** ***K44 - Knowledge of mobile application development, 2.67 average*** One employer noted that web development "is not the future." There are big implications for front-end engineers. Back-end engineers need to be aware of this as well. The apps in your smartphone will be an extension of the infrastructure. You will no longer visit websites. Another employer asked about the client-server technologies that might replace browser-server connections. The first employer explained that it's going to be "native applications written directly against Android and iOS." There are some hybrid rendering technologies as well, but those are geared more to legacy apps that started as websites. Are mobile apps going to just become user interfaces? Maybe not, said another employer. Smartphones have ML chips now that can do work that has traditionally been done on the server. Another employer noted that phones and apps now have secure storage, which means all of that data can reside on the phone. Mobile app architecture has to consider what will work in the absence of a network. Where will that functionality and data go? You can load a custom-trained LLM right onto a mobile device. In the example of a salesperson in the field using that

	LLM on the job, the only thing that app will do is "make a call home to verify open orders." Everything else will be local. ***K49 - Knowledge of applications with public keying by leveraging existing public key infrastructure (PKI) libraries and incorporating certificate management and encryption functionalities when appropriate, 2.50 average*** The BILT agreed that K49 should include PQA in addition to PKI. There was a discussion about removing the PKI mention – one employer argued that a new hire won't be brought in because of PKI knowledge; the company is going to want someone to help them get to PQA. This employer believed that the world is "one big PKI breach away" from PKI being gone for good. Another employer suggested rewording to cover communication encryption generally and make PKI just one example of that. No one objected. But one employer did note that security isn't just about logging on – every interaction is going to be authenticated and encrypted. It will permeate the architecture of everything. **ACTION: consider rewording K49 to "Knowledge of communication encryption, including PKI and PQA."** ***K50 - 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) when appropriate, 2.43 average*** **ACTION: consider including PQA as another example.**
Skills items	***S9 – Skill in using electronic mail software (e.g. Google Gmail; IBM Notes Hot technology; Microsoft Exchange Server Hot technology; Microsoft Outlook Hot technology), 2.92 average*** One employer suggested cutting this S item, suggesting that most people now text rather than email. Another employer countered that a lot of organizations still rely on email communications. They may be trying to move to portals, but that is not always possible. She argued that students need to know that email has not gone away for a lot of companies, which includes understanding that customer queries and problems often will still be communicated via email. ***S11 - 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.*** An educator in the chatbox asked about the value of C#. One employer wondered if it's better to talk about application versus system programming rather than focusing on the specific examples. There is also a blend of "what happens in the middle" at the user interface – things like React, Angular, and Swift. Another employer pointed out that React and Angular are frameworks, not languages. One employer agreed it's important to discuss applications programming versus systems programming and what's required for each. You have to consider memory, safety, security, and concurrency. He noted there are debates now about the usefulness and durability of Zig and Rust. In 36 months, it will all be different. Another employer noted that his clients all ask for Java and C# - those are foundational for enterprise; he doesn't see much C++. Another employer suggested the focus here should be more on problems the languages solve rather than the languages themselves. He has to jump across the stack so often that he uses all of the languages.

	S13 - Skill in using, incorporating and utilizing cookies, 2.31 average One employer noted that the important thing here is that students understand the security implications of cookies, especially with regard to laws like GDPR that specifically target cookies. ***S24 - Skill in producing technical content for tech writers, 2.31 average*** The BILT agreed this is not an entry-level skill. **ACTION: consider removing S24.**
What should be included next time	Ann next asked the employer what items were missing from the list and should be included for the next job skills vote. One employer mentioned PQA again, **which is linked to the concept of zero trust**. What does zero trust architecture look like when you start talking about post-quantum cryptography? What was supposedly a 10,000 year quantum protocol can now be broken in two weeks. The next step is "post quantum" which includes zero trust and decentralized identity management. How do I manage disparate IDs throughout an organization? This is where PQA fits into mobile development – an employee who's connected into the network with three different devices and you have to manage and secure all three. Multi-factor authentication is totally different on a phone than on a desktop. This concept is so new that he didn't think Gartner even had a name for it yet. From the chatbox: "Two-factor and multifactor authentication is huge; identity management solutions – huge." One employer suggested there's a big difference between where mobile development is today and where it's going to be. It's going to "shift to the right." Application development will not remain in this traditional space, so there's going to be a need to define a new skill set. he explained that today, mobile phones maintain a "wireless tether back to a tower somewhere and then back to a corporate network." In the future, however, **phones will be a truly mobile device be operating on its own** because of the intelligence being built in. The hardware is going to be totally different. This means that the evolution and development of the applications on those devices will need to be totally different. Whereas today, if you go into a shop of 20 developers, 4 don't know how to develop mobile, soon that shop of 20 will be split 50/50: 10 can develop apps and 10 can develop mobile. Another employer agreed that mobile development standards are becoming the standards for application development as well. One big company started using React for application development because that's what their front-end engineers understood. Shifting right in IT: https://medium.com/@ajay.monga73/the-dual-approach-maximizing-security-with-shift-left-and-shift-right-strategies-c096c048f845 Ann asked about "vibe coding" since earlier, one of the employers called it a media term and asked that it not be included. One of the other employers suggested "vibe coding" is better labelled **"AI assisted programming."** Another agreed that students won't get hired if they don't know prompt engineering. One employer noted that students will need to know the program specs in order to build the prompt. Another employer said this goes "beyond code generation." AI is impacting every task from design to review and daily management tasks like debugging. This skill is essential. Vibe coding: https://www.wired.com/story/vibe-coding-engineering-apocalypse/

	Another employer suggested the techniques of **system integration**, which include JSON, REST, and in some cases SOAP, and **data management/manipulation and identity management solutions,** which covers things like connecting to data sources, mining data to support metrics and reporting, and processing requests. These come up over and over for her clients. Another employer suggested that the overall concept of PQA and zero trust covers these. More and more, the industry is moving into things like verifiable credentials, where "every message is signed and it has an identity." He suggested technicians need to know all of these many techniques. PQA helps address the question of "what does it mean to be zero trus and how do you deal with that?" One employer wanted to talk more about **low code**. His company integrates low code with AI for efficiency and optimization, which helps eliminate a lot of the tedium and the redundancy that developers have to face. Another employer stressed the need for IT students and technicians to **embrace lifelong learning**. One employer added that students need skills for **developing APIs and team coding architecture for microservices**. The employers agreed that **security has to be "baked in from the very beginning."** It's not something to be added later. One employer explained that his expectation for a student graduating from a two-year program is that he/she can write good quality code that runs and is secure. What's required to make that happen will likely vary by employer, and entry-level workers will learn a lot on the job. The foundational skills of secure code-writing, however, whether with the help of AI or on your own, are essential. Another employer agreed, suggesting students need a lot of experience with a variety of simple application development. He likened it to learning how to play the piano. It takes practice. Another employer suggested that the classwork should probably be split between mobile apps and regular apps, with the understanding that in the future it's going to be mostly mobile-based. One employer responded that big companies are slow to move into mobile; many of them are still embracing – and updating – web applications. Another suggested that in the near term, it may come down to whether employees are working at a desk or in the field with a mobile device. This migration will take time. One employer summed it up: if students don't know mobile development, they're going to have a harder time finding a job. No company is now starting solely with web applications.
Conclusion	Mark announced the next two BILT meetings. A cross-disciplinary workforce trends meeting is set for Tuesday, August 19. Later in the fall, NITIC will host a job skills prioritization and discussion meeting on entry-level infrastructure and cloud job skills. Stephanie clarified that everyone attending today's meeting will get invited for the August 19 meeting. Outcomes from these BILT meetings (minutes, prioritization worksheets, and trends summaries) are available for free here: https://www.nitic.org/industry/national-bilt/bilt-overview/
Next Meeting: Tuesday, August 19 (9:30am-10:30am Central/10:30am-11:30am Eastern) – IT industry trends	