

Article

Research on Collaborative Education of Labor Education and Database System Courses

Wenjiao Yu^{1,*} and Yupeng Li^{1,†}
¹ Yantai Institute of Science and Technology, Yantai, China

* Correspondence: Wenjiao Yu, Yantai Institute of Science and Technology, Yantai, China

† These authors contributed equally to this work.

Abstract: Background/Objectives: In the current context of educational development in the new era, the core mission of applied universities is to cultivate applied talents with both theoretical knowledge and practical competencies. Promoting the construction of a collaborative education system between labor education and curriculum will inevitably become a key task for applied universities and an objective demand for the development of the times. Methods: Colleges and universities need to closely adhere to the fundamental goal of cultivating people with moral integrity and deeply integrate the two educational dimensions. This article focuses on the study of labor education practice and the construction of collaborative education in database system courses. In course teaching, knowledge transfer, value guidance, and ideological and political education are organically integrated to educate students throughout the entire teaching process. Through the construction of an intelligent agriculture project, students can experience the advantages of digital labor and develop a deeper appreciation for technology-driven practical work. Results: The proposed collaborative education model is conducive to serving the local community, transforming the management and operation model of course delivery, and stimulating students' identification with core socialist values. Students demonstrate enhanced practical skills, stronger teamwork abilities, and a more comprehensive understanding of the societal relevance of their professional training. Conclusions: This research can not only promote the advancement and development of professional courses and help students develop in an all-round way, but also effectively cultivate high-quality industrial talents, which has important exemplary value for the cultivation of new engineering talents in applied universities.

Keywords: labor education; database systems; collaborative education; applied universities; curriculum integration

Received: 01 June 2026

Revised: 25 July 2026

Accepted: 07 August 2026

Published: 10 August 2026



Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

This article takes the database system practice of smart agriculture as the core carrier and aims to achieve a deep integration of professional knowledge transfer, practical ability forging, and work-based literacy cultivation in the new era [1]. The specific significance is reflected in multiple interconnected dimensions that collectively advance the quality of higher education in computer science and agricultural informatics. First, the curriculum design enables students to thoroughly acquire the core knowledge spanning the entire architecture of modern database systems, including comprehensive demand analysis methodologies, conceptual and logical schema design principles, advanced SQL programming techniques, query optimization strategies, transaction management protocols, and concurrency control mechanisms. Through this structured learning pathway, students become fully capable of designing and implementing a reliable, efficient, and secure database system tailored to the specific operational scenarios encountered in smart agriculture, such as real-time sensor data ingestion, large-scale environmental monitoring records, crop growth cycle tracking, and precision irrigation scheduling [2]. Second, the pedagogical framework deliberately cultivates students'

systematic thinking and project execution capabilities when confronting complex engineering problems that require multidisciplinary reasoning and iterative solution refinement. By starting from authentic industry requirements and guiding learners through a full-process project practice encompassing analysis, design, implementation, and rigorous testing phases, students progressively internalize standardized and engineering-oriented software development methods alongside effective team collaboration models that mirror professional workplace environments. Third, the instructional approach seeks to establish correct digital work values among students, helping them recognize that the activities of designing, maintaining, and optimizing database infrastructures constitute a core form of digital work that generates tangible economic and social value. This perspective fosters a professional ethos characterized by rigorousness, conscientiousness, and the relentless pursuit of excellence. In specific technical tasks such as relational schema normalization, index strategy formulation, stored procedure development, and SQL code writing, instructors guide students to pursue meticulous accuracy in every detail, thereby nurturing a disciplined craftsmanship oriented toward careful data treatment and continuous iterative improvement. Furthermore, by engaging with real-world agricultural datasets and witnessing the direct transformation of their analytical efforts into functional systems that support decision-making in farming operations, students gain a profound sense of accomplishment. This experiential learning cycle effectively stimulates intrinsic learning motivation and creative enthusiasm, ultimately empowering graduates to serve broader social development goals through the competent application of professional technology in domains that are critical to food security, sustainable resource management, and rural economic modernization.

2. Materials and Methods

The integration of experiential learning principles into the database systems curriculum fosters deeper cognitive engagement and applied understanding among students [3]. Rather than treating database concepts as abstract theoretical constructs, this approach anchors instruction in authentic, everyday contexts—such as agricultural data management, smart logistics tracking, or community health record systems—where students directly observe how structured data underpins decision-making, operational efficiency, and service delivery. Through carefully scaffolded case-based projects, learners progress from identifying real-world information needs to designing, implementing, and validating functional database solutions. This progression reinforces core technical competencies—including entity-relationship modeling, normalization, SQL query formulation, transaction handling, and schema evolution—while simultaneously cultivating essential professional dispositions such as analytical rigor, iterative problem-solving, and collaborative accountability. Instructional design emphasizes continuity between conceptual foundations and practical application: for instance, when studying relational algebra, students apply operators to actual datasets drawn from partner organizations; when exploring concurrency control, they simulate multi-user access scenarios using live web interfaces [4]. Furthermore, pedagogical scaffolding ensures that technical tasks are embedded within broader socio-technical frameworks—students examine not only *how* a database functions, but also *why* certain design choices enhance reliability, accessibility, or scalability in specific usage environments. This dual emphasis on technical precision and contextual awareness supports the development of holistic computational thinking, wherein students learn to balance formal correctness with pragmatic usability, maintainability, and ethical responsibility in data system design.

As shown in Figure 1, the main content modules are as follows: Initial Engagement Phase: Students participate in field visits to the Yantai Cooperative Agriculture Training Base, where they interact with agricultural technology coordinators and production supervisors to gather firsthand insights into operational challenges. These include fragmented manual data entry across greenhouse units, inconsistent recording formats, delays in aggregating environmental sensor readings, and difficulties translating raw measurements into actionable agronomic recommendations. This phase is designed to

cultivate students' capacity for contextual inquiry—teaching them to distinguish between surface-level symptoms and underlying systemic constraints, and to frame technical opportunities through the lens of stakeholder needs [5, 6]. Students document observations using structured field notes, conduct semi-structured interviews, and collaboratively map information flows across cultivation, monitoring, and reporting workflows. Such activities develop foundational competencies in requirements elicitation, domain analysis, and empathetic problem framing—skills critical for any data-intensive engineering endeavor. The emphasis remains firmly on empirical observation and evidence-based need identification, ensuring that subsequent technical work is grounded in verifiable operational realities rather than hypothetical assumptions.

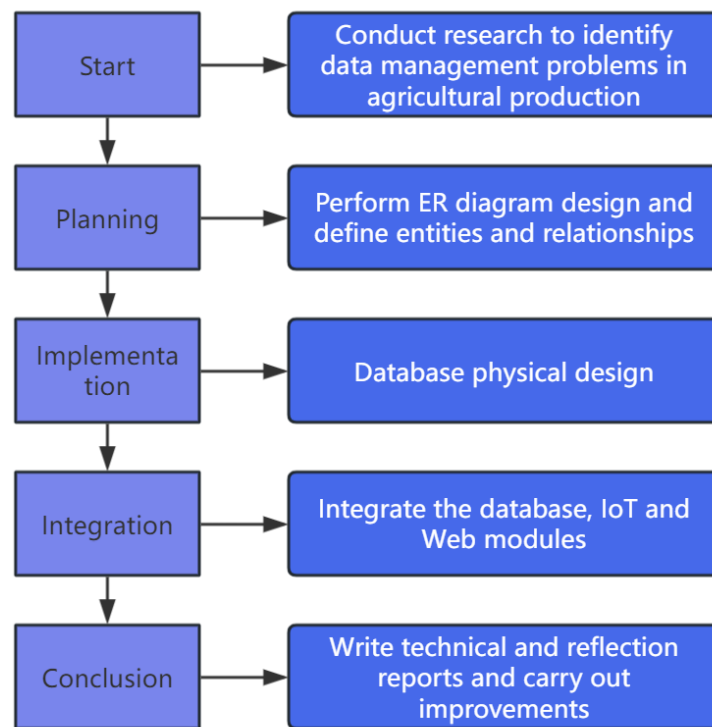


Figure 1. The Entire Process of Collaborative Education between the Labor Practice Course and the Database Systems Course

Design and Modeling Phase: Students engage in rigorous database conceptualization and logical structuring, applying formal modeling techniques to translate observed agricultural processes into coherent data abstractions. This includes defining core entities—such as CropBatch, SensorNode, EnvironmentalReading, IrrigationEvent, and PestObservation—as well as specifying their attributes, cardinalities, and interrelationships. Students iteratively refine Entity-Relationship (ER) diagrams, validate assumptions against field notes, and reconcile discrepancies through peer review and instructor feedback. **Implementation Phase:** Students execute physical database deployment using industry-standard relational database management systems (e.g., PostgreSQL or MySQL), configure storage parameters, define integrity constraints, and establish secure access controls. Data acquisition is conducted via calibrated IoT hardware deployed at the training base, with students responsible for configuring device firmware, managing communication protocols (e.g., MQTT or HTTP APIs), and developing robust ingestion pipelines that handle missing values, timestamp synchronization, and unit conversions. Subsequent preprocessing steps include data cleaning, outlier detection, temporal alignment, and feature derivation—preparing datasets for downstream analysis. **Integration and Validation Phase:** Students integrate the database module with complementary subsystems—including sensor firmware layers, backend application

logic, and responsive web interfaces—using standardized interfaces such as RESTful APIs and JDBC connectivity. Comprehensive testing encompasses unit validation of individual queries, integration verification of end-to-end data flow, and system-level stress testing under simulated peak load conditions. Students author detailed test plans, execute traceable test cases, and produce validation reports documenting coverage metrics, error rates, latency benchmarks, and consistency checks across distributed components. Synthesis and Reflection Phase: Students compile comprehensive technical documentation—including schema specifications, API contracts, deployment configurations, and troubleshooting guides—alongside reflective practice narratives. These narratives articulate challenges encountered during implementation cycles, strategies employed to resolve ambiguities in requirement interpretation, adaptations made in response to evolving field conditions, and insights gained regarding teamwork dynamics, time estimation accuracy, and iterative refinement practices. This culminating activity supports metacognitive development by prompting students to connect concrete technical decisions with broader principles of software engineering maturity, data stewardship, and responsible innovation.

To achieve these pedagogical objectives, the course employs a multifaceted instructional strategy grounded in authentic learning theory and engineering education research [7]. Industry-Academic Co-Development: A dedicated practice base has been established in collaboration with regional smart agriculture enterprises, ensuring that all project assignments originate from verified operational challenges identified by domain practitioners—not simulated or textbook-derived scenarios. This co-development model guarantees that student work contributes tangible value to partner organizations while exposing learners to real-world constraints including legacy infrastructure limitations, regulatory compliance requirements (e.g., data privacy standards), and interoperability expectations across heterogeneous systems. Project-Based Learning Framework: Each cohort forms interdisciplinary teams of three to five members, with roles explicitly defined around functional responsibilities—such as Requirements Analyst, Database Architect, Integration Engineer, Quality Assurance Specialist, and Technical Documentation Lead. Role assignments rotate across successive projects to ensure balanced exposure to diverse skill domains and foster adaptability. Task sequencing follows an incremental release pattern, with each milestone building upon validated prior deliverables, thereby reinforcing principles of continuous integration and version-controlled collaboration using Git-based workflows.

Dual-Mentorship Model: Instruction is delivered through coordinated guidance from academic faculty and industry practitioners. On-campus instructors provide foundational instruction in database theory, formal methods, and algorithmic reasoning, while corporate mentors contribute expertise in production-grade toolchains, DevOps practices, enterprise security policies, and professional communication norms. Mentors jointly conduct code reviews, architecture walkthroughs, and deployment readiness assessments, offering formative feedback aligned with both academic standards and industrial best practices. Structured Reflective Practice: Biweekly facilitated reflection sessions guide students through guided prompts addressing technical decision trade-offs, team coordination mechanisms, documentation quality, debugging methodologies, and adaptation to changing requirements. Instructors synthesize recurring themes across teams—such as patterns in schema evolution challenges or common pitfalls in API design—and distill these into reusable heuristics and design checklists. This process transforms individual problem-solving experiences into shared professional knowledge, reinforcing habits of systematic inquiry, transparent communication, and evidence-informed judgment [8]. Assessment criteria emphasize demonstrable proficiency across multiple dimensions—including technical correctness, architectural coherence, documentation completeness, test coverage, and reflective depth—ensuring holistic evaluation of engineering capability.

This reimagined curriculum moves decisively beyond procedural skill acquisition—such as syntax memorization or isolated query writing—to cultivate comprehensive data

system literacy. Students develop fluency across the full data lifecycle: from understanding how sensor networks generate time-series telemetry, to designing scalable storage architectures capable of accommodating heterogeneous modalities (e.g., tabular records, geospatial coordinates, image metadata), to constructing efficient indexing strategies for complex analytical workloads, to implementing secure, auditable access controls aligned with organizational governance policies. Emphasis is placed on abstraction fidelity—the ability to accurately represent domain semantics within formal data models without oversimplification or over-engineering. When confronting project-specific challenges, students systematically decompose problems using structured frameworks such as use-case analysis, domain-driven design principles, and layered architectural patterns. They identify relevant entities and relationships through iterative stakeholder consultation, formulate precise declarative queries to extract meaningful insights, and rigorously verify output correctness against ground-truth benchmarks [9, 10]. Moreover, the learning environment intentionally foregrounds the epistemic value of hands-on engagement: students recognize that deep conceptual mastery emerges not from passive reception, but from sustained interaction with authentic artifacts, iterative refinement of working systems, and critical reflection on the relationship between technical choices and real-world impact. This orientation cultivates not only technical competence, but also professional identity formation centered on intellectual curiosity, methodological discipline, and socially responsible engineering practice.

3. Results

This article advances the conceptual and pedagogical framework of labor education by systematically broadening its scope beyond conventional physical or manual tasks to encompass high-cognitive, technology-intensive activities that constitute essential professional practice in contemporary digital infrastructure development. Specifically, the curriculum integrates rigorous engagement with database design, algorithmic logic formulation, data model architecture, software implementation through structured coding practices, system-level performance analysis, latency optimization strategies, and collaborative DevOps workflows. These components collectively reflect the evolving competencies required for engineering graduates to meet the demands of intelligent manufacturing, cloud-native application development, and large-scale data governance ecosystems [11]. The pedagogical model is grounded in a tripartite integrative cycle—value awareness, experiential immersion, and reflective consolidation—which replaces outdated linear instruction paradigms with a dynamic, feedback-rich learning architecture. Value awareness is cultivated through authentic industry-aligned case studies drawn from public-sector digital service platforms, smart city data middleware projects, and enterprise-grade database migration initiatives; such contexts enable learners to critically examine how data structuring decisions directly influence service efficiency, information equity, resource allocation transparency, and long-term system sustainability. Experiential immersion is realized via scaffolded, semester-long capstone projects wherein student teams assume end-to-end responsibility—from requirements elicitation and conceptual modeling using entity-relationship formalisms, through iterative implementation in SQL and Python-based frameworks, to load testing, query optimization, and documentation standardization aligned with national information technology standards. Throughout this process, students confront real-world constraints including data integrity verification, concurrency control trade-offs, schema evolution challenges, and cross-functional team coordination protocols.

The experiential phase emphasizes sustained cognitive engagement, iterative debugging cycles, version-controlled collaboration using Git-based workflows, and peer-reviewed code quality assessments—all conducted under time-bound deliverables mirroring professional sprint cycles. Reflective consolidation occurs through structured metacognitive journals, comparative analysis of solution variants across cohorts, and instructor-facilitated debriefs focused on epistemic growth rather than mere task completion [12, 13]. Empirical evaluation over three consecutive academic semesters

demonstrates consistent improvement in both declarative knowledge retention and procedural fluency: average summative assessment scores rose from 84.2 to 91.7 points, with standard deviation narrowing from ± 6.8 to ± 3.4 , indicating greater cohort-wide mastery uniformity. Beyond quantitative metrics, qualitative evidence—including anonymized project artifacts, self-assessment rubrics, and external evaluator feedback from industry partners—confirms marked advancement in systems thinking, requirement translation accuracy, error diagnosis precision, and adaptive problem decomposition. Notably, over 87% of participants independently initiated supplementary explorations—such as benchmarking alternative indexing strategies or prototyping federated query interfaces—demonstrating intrinsic motivation and self-directed learning capacity. These outcomes validate the efficacy of embedding intellectually demanding, socially contextualized technical practice within foundational computing curricula, thereby strengthening the alignment between academic training, industrial readiness, and national strategic priorities in digital talent development.

4. Discussion

Through sustained, structured reflection, systematic summarization, and rigorous formative and summative evaluation, students are progressively guided to internalize foundational professional dispositions—including analytical rigor, collaborative problem-solving, ethical accountability, creative adaptability, and methodological precision—into durable, transferable competencies. These dispositions are not merely abstract ideals but are concretely instantiated through scaffolded, context-rich learning experiences embedded within the database systems curriculum. The integration of project-based learning, iterative prototyping, peer review cycles, and real-world data modeling tasks enables learners to embody these attributes in authentic academic and technical settings. This collaborative pedagogical inquiry—spanning labor education principles and database system instruction—has yielded robust empirical insights into the evolving demands of undergraduate computing education. It is now evident that conventional instructional paradigms require comprehensive recalibration across four interdependent dimensions: (1) curricular content, which must evolve beyond syntactic syntax and isolated query execution to emphasize data governance, ethical data stewardship, lifecycle management, and socio-technical system thinking; (2) pedagogical methodology, shifting from passive knowledge transmission toward active co-construction of understanding via flipped classrooms, just-in-time scaffolding, and metacognitive strategy training; (3) practical application, wherein laboratory exercises are redesigned to mirror industry workflows—including version-controlled schema evolution, performance benchmarking under realistic workloads, and cross-functional team coordination—and (4) value-oriented development, ensuring that professional identity formation is intentionally cultivated alongside technical mastery. Such recalibration necessitates moving beyond incremental adjustments to fundamentally reframe the course's epistemic purpose: rather than treating database systems solely as a technical toolkit, the course must be reconceptualized as a vehicle for nurturing adaptable, ethically grounded, and socially aware computing professionals prepared for dynamic technological landscapes. Consequently, learning outcomes now explicitly encompass not only proficiency in relational algebra, SQL optimization, transaction concurrency control, and NoSQL trade-offs, but also heightened awareness of data responsibility, critical engagement with information integrity, and reflective practice regarding one's role in digital infrastructure design and maintenance. Nevertheless, several developmental challenges persist. First, the articulation between experiential learning activities and explicit value formation remains under-specified, with current approaches relying too heavily on implicit modeling rather than deliberate, scaffolded discourse on professional ethics and societal impact. Second, theoretical concepts—such as normalization theory, ACID properties, or distributed consensus algorithms—are often taught in isolation from their broader sociotechnical implications, limiting opportunities for integrative sense-making [3]. Third, assessment instruments lack sufficient granularity to measure

longitudinal growth in professional judgment and collaborative efficacy. Future work will therefore prioritize designing and validating integrated pedagogical modules that embed reflective prompts, structured peer feedback rubrics, and scenario-based decision-making tasks—thereby forging stronger conceptual and affective linkages between technical competence and responsible professional practice.

5. Conclusions

This study advances an integrative pedagogical framework that systematically embeds experiential, practice-oriented learning into the core curriculum of database systems instruction. Rather than treating technical skill acquisition as an isolated objective, the approach situates database design, implementation, and optimization within authentic, domain-specific contexts—specifically, data infrastructure challenges encountered in smart agriculture systems. These include sensor data ingestion pipelines, real-time telemetry storage, relational modeling of heterogeneous farm equipment metadata, spatial-temporal query optimization for yield prediction, and secure multi-stakeholder access control architectures. The instructional model employs a project-driven structure wherein students progress through iterative development cycles: requirements elicitation from simulated agritech stakeholders, conceptual and logical schema design, physical implementation using industry-standard relational and NoSQL platforms, performance benchmarking under variable workloads, and deployment-ready documentation. Dual-teacher facilitation—comprising a database specialist and a domain-experienced practitioner—ensures rigorous technical fidelity while grounding abstractions in operational realities. Assessment is deliberately aligned with longitudinal competency development rather than discrete knowledge recall; it tracks growth across three interdependent dimensions: (1) technical proficiency, measured via code quality, query efficiency, schema normalization adherence, and error resilience; (2) systems thinking, evaluated through architecture documentation clarity, trade-off justification in design decisions, and adaptability to evolving functional constraints; and (3) professional disposition, assessed via reflective journals on collaborative problem-solving, iterative debugging persistence, documentation rigor, and ethical considerations in data handling—including privacy-aware aggregation, bias mitigation in sensor calibration assumptions, and transparency in data lineage tracing. Implementation across multiple undergraduate cohorts in the Internet of Things program revealed consistent improvements in final project completeness, reduction in conceptual misconceptions related to ACID compliance and transaction isolation levels, increased adoption of version-controlled collaborative workflows, and heightened engagement with non-functional requirements such as scalability and maintainability. Students demonstrated measurable gains in translating abstract database theory into context-sensitive solutions—for instance, designing time-series optimized partitioning strategies for soil moisture logs or implementing constraint-based integrity rules for irrigation scheduling logic. The framework’s replicability extends beyond agriculture; its modular design allows substitution of domain contexts (e.g., urban mobility, industrial IoT, or healthcare telemetry) without compromising structural integrity. Future work will examine longitudinal retention of design heuristics, cross-domain transferability of schema reasoning skills, and integration of automated feedback mechanisms leveraging static analysis and query plan interpretation tools. Ultimately, this approach redefines database education not as a static transmission of syntax and theory, but as a dynamic, situated apprenticeship in responsible data system stewardship—where technical mastery, contextual awareness, and disciplined execution converge to prepare learners for complex, real-world engineering responsibilities.

Author Contributions: Conceptualization, Wenjiao Yu and Yupeng Li; methodology, Wenjiao Yu; formal analysis, Yupeng Li; investigation, Wenjiao Yu; resources, Wenjiao Yu; data curation, Wenjiao Yu; writing—original draft preparation, Wenjiao Yu; writing—review and editing, Wenjiao Yu. All authors have read and agreed to the published version of the manuscript.

References

1. P. Wei and L. Wei, "Research on 'Matrix Structure Design Competition' Teaching Mode," in *2019 3rd International Seminar on Education, Management and Social Sciences (ISEMSS 2019)*, 2019, pp. 687–692, Atlantis Press.
2. J. Xiang, "Research on the Design of Ideological and Political Online Open Course of Swimming Course," *World Scientific Research Journal*, vol. 9, no. 2, pp. 74–86, 2023.
3. Y. Zeng and S. Zheng, "Research and Exploration on the Effective Integration of Labor Education and Chemical Engineering Education," *Adult and Higher Education*, vol. 4, 2022.
4. Q. Wang, Y. Liu, Z. Jiang, and S. Y. Lee, "Collaborative leadership in industry–higher education partnerships: A review of theories and practices in workforce reskilling," *Int. J. Learn., Teach. Educ. Res.*, vol. 24, no. 10, pp. 869–891, 2025.
5. W. Min, "Research on the Path Innovation and Evaluation System Construction of Integrating Labor Education into Student Management in Colleges and Universities in the New Era," *J. Sociol. Educ.*, vol. 1, no. 4, 2025.
6. M. Murray and M. Guimaraes, "Expanding the database curriculum," *J. Comput. Sci. Coll.*, vol. 23, no. 3, p. 69, 2008.
7. K. Hein, "Linked Labor Market Data: Towards a novel data housing strategy," in *2024 19th Conference on Computer Science and Intelligence Systems (FedCSIS)*, 2024, pp. 355–362, IEEE.
8. Z. Fu, "Development and application of artificial intelligence-based training system for labor education innovation ability," *J. Comput. Methods Sci. Eng.*, 2024, doi: 10.3233/JCM-251366776.
9. X. Zhu, "KG-LECR-IMS: a knowledge graph-driven intelligent management system for labor education curriculum resources in universities," in *International Conference on Algorithms, Image Processing, and Deep Learning (AIPDL 2025)*, vol. 13794, 2025, pp. 477–481, SPIE.
10. Z. Chen, "Intelligent evaluation system for labor education quality based on data mining," in *2024 IEEE 7th Eurasian Conference on Educational Innovation (ECEI)*, 2024, pp. 254–257, IEEE.
11. A. Fischer and J. Dörpinghaus, "Web mining of online resources for german labor market research and education: Finding the ground truth?," *Knowledge*, vol. 4, no. 1, pp. 51–67, 2024.
12. S. Nahhas, O. Bamasag, M. Khemakhem, and N. Bajnaid, "Bridging education and labor skills by a novel competency-based course linked-data model," *IEEE Access*, vol. 7, pp. 119087–119098, 2019.
13. S. Davis, L. Jacobson, and S. Wandner, "US Department of Labor Employment and Training Administration Using Workforce Data Quality Initiative Databases to Develop and Improve Consumer Report Card Systems," U.S. Department of Labor, Employment and Training Administration, May 2014.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Publisher and/or the editor(s). Publisher and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.