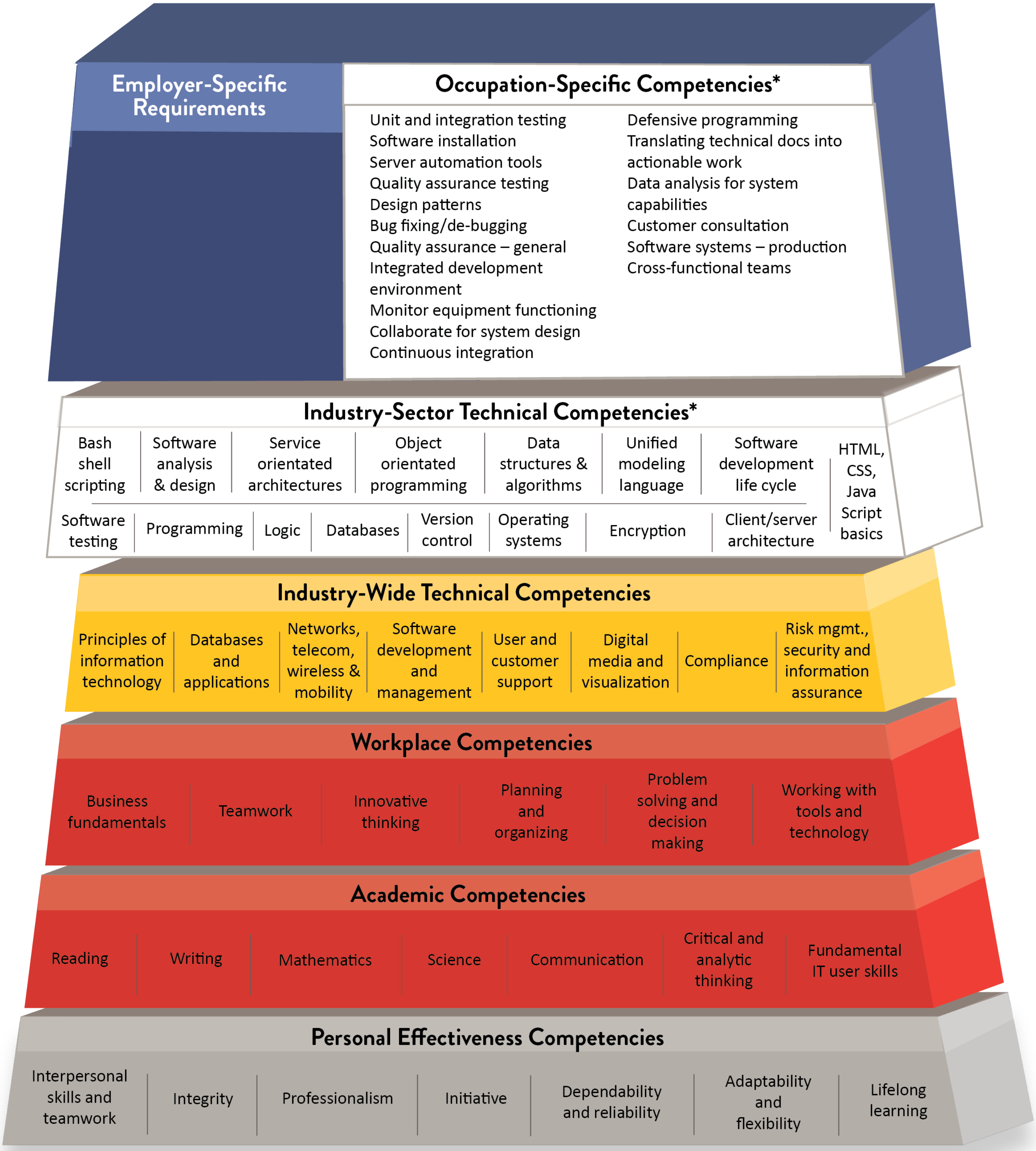
Minnesota Dual-Training Pipeline

Competency Model for Information Technology
Occupation: Web Developer - Back End



Employer-Specific Requirements

Occupation-Specific Competencies*

- Unit and integration testing
- Software installation
- Server automation tools
- Quality assurance testing
- Design patterns
- Bug fixing/de-bugging
- Quality assurance – general
- Integrated development environment
- Monitor equipment functioning
- Collaborate for system design
- Continuous integration
- Defensive programming
- Translating technical docs into actionable work
- Data analysis for system capabilities
- Customer consultation
- Software systems – production
- Cross-functional teams

Industry-Sector Technical Competencies*

- Bash shell scripting
- Software analysis & design
- Service orientated architectures
- Object orientated programming
- Data structures & algorithms
- Unified modeling language
- Software development life cycle
- HTML, CSS, Java Script basics
- Software testing
- Programming
- Logic
- Databases
- Version control
- Operating systems
- Encryption
- Client/server architecture

Industry-Wide Technical Competencies

- Principles of information technology
- Databases and applications
- Networks, telecom, wireless & mobility
- Software development and management
- User and customer support
- Digital media and visualization
- Compliance
- Risk mgmt., security and information assurance

Workplace Competencies

- Business fundamentals
- Teamwork
- Innovative thinking
- Planning and organizing
- Problem solving and decision making
- Working with tools and technology

Academic Competencies

- Reading
- Writing
- Mathematics
- Science
- Communication
- Critical and analytic thinking
- Fundamental IT user skills

Personal Effectiveness Competencies

- Interpersonal skills and teamwork
- Integrity
- Professionalism
- Initiative
- Dependability and reliability
- Adaptability and flexibility
- Lifelong learning

Based on: Information Technology Competency Model Employment and Training Administration, United States Department of Labor, September 2012. *Pipeline recommends the Industry-Sector Technical Competencies as formal training opportunities (provided through related instruction) and the Occupation-Specific Competencies as on-the-job training opportunities.

Competency Model for Web Developer – Back End

Web Developer Back End – A Web Developer-Back End is one who specializes in the development of websites and webpages, primarily the behind-the-scenes coding and programming that is required for creating a fully functioning website.

Industry-Sector Technical Competencies

Related Instruction for dual training means the organized and systematic form of education resulting in the enhancement of skills and competencies related to the dual trainee's current or intended occupation.

- **Bash shell scripting** – Know how to script a UNIX shell or command language.

- **Software testing** – Know how to evaluate software to make sure it meets specific requirements. Also know how to identify any gaps, errors, or missing requirements for website development.

- **Software analysis and design** – Understanding of modeling and its central role in eliciting, understanding, analyzing, and communicating software requirements, architecture, and design for website development.

- **Programming** – Understand how to create programs by writing "code" in a programming language.

- **Service oriented architectures** – Understand the architectural pattern in computer software design in which application components provide services to other components via a communications protocol, typically over a network.

- **Logic** – Understand the part of the program that encodes the real-world business rules that determine how data can be created, displayed, stored, and changed.

- **Object oriented programming** – Understanding this type of programming in which programmers define not only the data type of a data structure, but also the types of operations (functions) that can be applied to the data structure.

- **Databases** – Knowledge of implementing data models and database designs to ensure security and data integrity in database software operating for the website.

- **Version control** – Understanding of the system that records changes to a file or set of files over time so that you can recall specific versions later.

- **Data structures and algorithms** – Knowledge of the use of data structures and algorithms in software programming for web design.

- **Operating systems** – Understand the function of operating systems and how to properly create websites to interact with them.

- **Unified modeling language** – Understanding of the general-purpose modeling language for website engineering, designed to provide a standard way to visualize the design of a system.

- **Encryption** – Understanding of how encryption functions and how to work with it within the website development environment.

- **Software development life cycle** – Knowledge of Waterfall and Agile approaches to software development and when to use the appropriate model for website development.

- **Client/server architecture** – Knowledge of the Client/Server Architecture model and how to develop websites for such a system.

- **HTML, CSS, Java Script basics** – Knowledge of the common formatting and programming languages – HTML, CSS, JavaScript.

Occupation-Specific Competencies

On-the-Job Training (OJT) is hands-on instruction completed at work to learn the core competencies necessary to succeed in an occupation. Common types of OJT include job shadowing, mentorship, cohort-based training, assignment-based project evaluation and discussion-based training.

- **Unit and integration testing** – Be able to test various computing scenarios for units and integration.

- **Software installation** – Understand how to assist with software installation for the organization and individual user.

- **Server automation tools** – Know how to use applications which automate computing functions.

- **Quality assurance testing**– Know how to run tests on software and test for compatibility and functionality issues for the website.

- **Design patterns** – Understand how to learn and develop design patterns for problem solving in programming.

- **Bug fixing/de-bugging** – Know how to locate, fix, or bypass errors (bugs) in code or devices.

- **Quality assurance- general** – Be able to use appropriate methods to verify overall quality of website design and systems work properly.

- **Integrated development environment** – Know how to use the IDE application for website development.

- **Monitor equipment functioning** – Understand how to monitor system and review information from system to detect or assess problems.

- **Continuous integration** – Be able to merge developer working copies with a shared mainline several times a day.

- **Collaborate for system design** – Ability to collaborate with the development team which may include systems analysts, engineers and programmers.

- **Translating technical documents into actionable work** – Understand how to create working and actionable process documents from very technical IT documents.

- **Data analysis for system capabilities** – Know how to store, retrieve and manipulate data for analysis of system capabilities and requirements.

- **Customer consultation** – Know how to work with internal and external customers to gather information regarding system design, performance, and maintenance.

- **Software systems production** – Demonstrate ability to design, develop and modify software systems to run the website.

- **Defensive programming** – Ability to design model intended to ensure the continuing function of a website under unforeseen circumstances.

- **Cross-functional teams** – Understand how the web development role intersects with working with cross-functional teams in the organization.

Updated July 2022