CompuScholar, Inc.

Alignment to Ohio "**Web Design**" Course Standards

Ohio Course Details:

Course Title:	Information Technology
Course Code(s):	145010 - Web Design
Credit:	1
Grade Level:	6th-12th
State Standards Link:	http://education.ohio.gov/Topics/Career-Tech/Information-Technology-Career-Field

CompuScholar Course Details:

Course Title:	Web Design
Course ISBN:	978-0-9887070-3-0
Course Year:	2019

Note 1: Citation(s) listed may represent a subset of the instances where objectives are met throughout the course.

Note 2: Citation(s) for a "Lesson" refer to the "Lesson Text" elements and associated "Activities" within the course, unless otherwise noted. The "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts, and the Lesson Text contains full details.

Course Description

Students will learn the dynamics of the Web environment while pursuing an in-depth study of both Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS). Web based protocols such as FTP, TCP/IP, and HTTP will be addressed. Students will create a website with tag text elements, special characters, lines, graphics, hypertext links, and graphical tables.

Course Standards

Strand 2 - IT Fundamentals	
Learners apply fundamental principles of IT, including the history of IT and its impact on society, common industry terms, systems theory, information storage and retrieval, database management, and computer hardware, software, and peripheral device configuration and installation. This base of knowledge and skills may be applied across the career field.	
2.4. Emerging Technologies: Identify trending technologies, their fundamental architecture, and their value in the marketplace.	**CITATION(S)**
2.4.1. Investigate the scope and the impact of mobile computing environments on society.	N/A
2.4.2. Describe the differences, advantages, and limitations of cloud computing (e.g., public cloud, private cloud, hybrid cloud) and on premises computing.	N/A

2.4.3. Utilize cloud computing applications (e.g. services, applications, virtual environments).	Students use a cloud-based LMS throughout the course
2.4.4. Describe emerging technologies (e.g., Bring your Own Device [BYOD], Services Virtualization, Augmented Reality [AR], SMART Devices, Additive Manufacturing [3D Printing]).	Chapter 16, Lesson 3
2.7. Web Architecture: Explain the fundamentals of delivering information and applications using web architecture.	**CITATION(S)**
2.7.1. Describe methods of securely transmitting data.	Chapter 16, Lesson 2
2.7.2. Describe ways to present data (e.g., mobile applications, desktop applications, web applications).	Chapter 10, Lesson 1
2.7.3. Differentiate between a client and a server.	Chapter 13, Lesson 3
2.7.4. Identify how the use of different browsers and devices affects the look of a webpage.	Chapter 1, Lesson 1 Chapter 20, Lesson 1
2.7.5. Explain the relationship between data transmission volumes, bandwidth, and latency.	Supplemental Lesson 1
2.7.6. Describe the characteristics and use of browser plug-ins.	Chapter 19, Lesson 1 Supplemental Lesson 6
2.7.7. Compare the advantages and disadvantages of running an in-house server or using a service provider.	Supplemental Lesson 1
2.7.8. Describe the difference between static and dynamic sites and the reasons for using each.	
2.9. Project Concept Proposal: Develop a project concept proposal.	**CITATION(S)**
2.9.1. Identify and incorporate branding strategies.	Chapter 10
2.9.2. Determine the scope and purpose of the project.	Chapters 14, 28 Supplemental Lesson 4
2.9.3. Determine the target audience, client needs, expected outcomes, objectives, and budget.	Chapters 14, 28 Supplemental Lesson 4
2.9.4. Develop a conceptual model and design brief for the project.	Chapter 13, Lesson 1 Chapters 14, 28
2.9.5. Develop a timeline, communication plan, task breakdown, costs (e.g., equipment, labor), deliverables, and responsibilities for completion.	Chapters 14, 28 Supplemental Lesson 4
2.9.6. Develop and present a comprehensive proposal to stakeholders.	Chapters 14, 28 Supplemental Lesson 4
2.11. Troubleshooting: Select and apply troubleshooting methodologies for problem solving.	**CITATION(S)**
2.11.1. Identify the problem.	Chapter 3, Lesson 2 Chapter 25, Lesson 3 Chapter 26, Lesson 3
2.11.2. Select troubleshooting methodology (e.g., top down, bottom up, follow the path, spot the differences).	Chapter 3, Lesson 2 Chapter 25, Lesson 3

2.11.3. Investigate symptoms based on the selected methodology.	Chapter 3, Lesson 2 Chapter 25, Lesson 3
2.11.4. Gather and analyze data about the problem.	Chapter 3, Lesson 2 Chapter 25, Lesson 3
2.11.5. Design a solution.	Chapter 14, Lesson 1 Chapter 28
2.11.6. Test a solution.	Chapter 14, Lesson 3 Chapter 28
2.11.7. Implement a solution.	Chapter 14, Lesson 2 Chapter 28
2.11.8. Document the problem and the verified solution.	Chapter 14, Lesson 3 Chapter 28
2.12. Performance Tests and Acceptance Plans: Develop performance tests and acceptance plans.	**CITATION(S)**
2.12.1. Create a written procedure agreed by the stakeholders and project team for determining the acceptability of the project deliverables.	Chapter 14, Lesson 1 Chapter 28 Supplemental Lesson 4
2.12.2. Develop a test system that accurately mimics external interfaces.	N/A
2.12.3. Develop test cases that are realistic, compare with expected performance, and include targeted platforms and device types.	Chapter 14, Lesson 3 Chapter 28
2.12.4. Develop, perform, and document usability and testing integration.	Chapter 14, Lesson 3 Chapter 28
2.12.5. Make corrections indicated by test results.	Chapter 14, Lesson 3 Chapter 28
2.12.6. Seek stakeholder acceptance upon successful completion of the test plan.	Chapter 14, Lesson 3 Chapter 28
2.13. Rollout and Handoff: Plan rollout and facilitate handoff to customer.	**CITATION(S)**
2.13.1. Include overall project goals and timelines in the rollout plan.	Chapter 14, Lesson 1 Chapter 28
2.13.2. Communicate rollout plans to key stakeholders in a timely manner.	Chapter 14, Lesson 1 Chapter 28
2.13.3. Conduct final review and approvals according to company standards.	Chapter 14, Lesson 3 Chapter 28
2.13.4. Identify support staff, training needs, and contingency plans in the rollout plan.	Supplemental Lesson 4
2.13.5. Test delivered application to assure that it is fully functional for the customer or user and meets all requirements.	Chapter 14, Lesson 3 Chapter 28
2.13.6. Deliver support and training materials.	N/A

Strand 4 - Infrastructure Systems	
Learners apply principles of networking and infrastructure related to the installation, administration, and maintenance of computer networks and components. Knowledge and skills may be applied to network connectivity, cabling, protocols, architecture, classification, topologies, operating systems, Open Systems Interconnection (OSI) standards, data encoding, Quality of Service (QoS), Internet Protocol (IP) addressing, and wide area network (WAN) design.	
4.6. Network Protocols: Compare network protocols.	**CITATION(S)**
4.6.1. Explain network protocols (e.g., Transmission Control Protocol/Internet Protocol [TCP/IP], User Datagram Protocol [UDP], Internet Protocol Version 4 [IPv4], Internet Protocol Version 6 [IPv6]).	Chapter 15, Lesson 2 Chapter 15, Lesson 3
4.6.2. Identify the advantages and disadvantages of well-known protocols (e.g., Domain Name System [DNS], File Transfer Protocol [FTP], Secure Hypertext Transfer Protocol [HTTPS], Telecommunications Network [Telnet], Dynamic Host Configuration Protocol [DHCP], Remote Desktop Protocol [RDP]) and associated port numbers.	Chapter 15, Lesson 3 Chapter 15, Lesson 5 Chapter 16, Lesson 2
4.6.3. Explain the purposes of encapsulation and decapsulation and their relationship to the Open Systems Interconnection (OSI) model.	N/A
4.6.4. Explain the difference between User Datagram Protocol (UDP) and Transmission Control Protocol (TCP).	N/A
4.6.5. Identify Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) conventional ports (e.g., Simple Mail Transfer Protocol [SMTP], Telnet, Hypertext Transfer Protocol [HTTP], File Transfer Protocol [FTP]).	Chapter 15, Lesson 2 Chapter 15, Lesson 5
4.6.6. Explain Transmission Control Protocol/Internet Protocol (TCP/IP) protocol details (e.g., Internet addresses, Address Resolution Protocol [ARP], Reverse Address Resolution Protocol [RARP], IP datagram format, routing IP datagrams, TCP segment format, IPv4, IPv6).	Chapter 15, Lesson 2 Chapter 15, Lesson 3
4.6.7. Describe a Virtual Private Network (VPN) and identify associated protocols (e.g., Layer 2 Tunneling Protocol [L2TP], Point-to-Point Tunneling Protocol [PPTP]).	Chapter 15, Lesson 4
4.6.8. Capture and analyze data packets.	N/A
4.7. Transmission Control Protocol/Internet Protocol (TCP/IP): Describe IP addressing schemes and create subnet masks.	**CITATION(S)**
4.7.1. Explain Fully Qualified Domain Names (FQDNs) and how they are used.	Chapter 15, Lesson 3
4.7.2. Explain the IP addressing scheme and how it is used.	Chapter 15, Lesson 3
4.7.3. Identify Class A, B, and C reserved (i.e., private) address ranges and why they are used.	N/A
4.7.4. Identify the class of network to which a given address belongs.	N/A
4.7.5. Differentiate between default subnet masks and custom subnet masks.	N/A

4.7.6. Explain the relationship between an IP address and its associated subnet mask.	N/A
4.7.7. Identify the differences between classful and classless addressing schemes.	N/A
4.7.8. Identify multicasting addresses and explain why they are used.	N/A
4.7.9. Create custom subnet masks to meet network design requirements.	N/A
4.7.10. Compare and contrast Internet Protocol Version 4 (IPv4) and Internet Protocol Version 6 (IPv6).	Chapter 15, Lesson 3

Strand 6 - Web Development	
Learners apply principles of design and technology, including programming standards and protocols, to create, test, host, and maintain webpages and websites with text, graphics, multimedia, scripting, linking, and data integration in a structure that is easy to navigate and accessible for all users via a variety of hardware and software platforms.	
6.1. Webpages: Create basic webpages.	**CITATION(S)**
6.1.1. Describe the basic principles of Hypertext Markup Language (HTML) and its functional relationship with web browsers.	Chapter 1, Lesson 1 Chapter 1, Lesson 2
6.1.2. Plan a webpage considering subject, devices, audience, layout, color, links, graphics, and Americans with Disabilities Act (ADA) requirements.	Chapter 14 Chapter 28
6.1.3. Format the text of a webpage in a WYSIWYG (What You See Is What You Get) editor and in a text editor using HTML formatting tags (e.g., hyperlink, e-mail, table formatting, graphic attributes).	Chapter 1 - 14: text editors Chapter 15 - 28:
6.1.4. Use writing process techniques (i.e., drafting, revising, editing, proofreading) to check the webpage for format and text accuracy.	Chapter 14 Chapter 28
6.1.5. Create and format ordered and unordered lists on a webpage using HTML list formatting tags.	Chapter 4, Lesson 3
6.1.6. Create and format a table in a webpage using HTML table formatting tags and attributes.	Chapter 12
6.1.7. Integrate styles (e.g., inline or external Cascading Style Sheets [CSS]).	Chapter 6
6.2. Links and Multimedia: Add links to a webpage and insert multimedia files.	**CITATION(S)**
6.2.1. Create absolute links and relative links.	Chapter 5, Lesson 2 Chapter 5, Lesson 4
6.2.2. Write a Hypertext Markup Language (HTML) anchor that links to another section of the same webpage.	Chapter 5, Lesson 1
6.2.3. Create hyperlinks that send e-mail messages and download files.	Chapter 5, Lesson 5 Chapter 19, Lesson 1
6.2.4. Insert image and wrap text around the image using Cascading Style Sheets (CSS).	Chapter 11, Lesson 2 Chapter 11, Lesson 3
6.2.5. Resize a graphic image in a webpage using CSS.	Chapter 11, Lesson 2

6.2.6. Insert media files (e.g., audio, video,) into a web page using HTML tags.	Chapter 19
6.2.7. Build a hover or mouseover effect to change the style of a link.	Chapter 8, Lesson 1
6.3. Scripting: Integrate scripting into a webpage.	**CITATION(S)**
6.3.1. Select and apply scripting languages used in web development.	Chapters 25, 26, 27
6.3.2. Insert client-side script into a webpage.	Chapters 25, 26, 27
6.3.3. Insert comments into client-side scripts.	Chapter 25, Lesson 1
6.4. Web Forms: Integrate forms into a webpage.	**CITATION(S)**
6.4.1. Design a data entry form from specifications that will accept a variety of user inputs (e.g., radio buttons, text entry fields, check boxes, drop-down menus).	Supplemental Lesson 11
6.4.2. Write the Hypertext Markup Language (HTML) code to add a form to a webpage.	Supplemental Lesson 11
6.4.3. Write the HTML code to add text entry fields, radio buttons, check boxes, drop-down menus, and other user inputs to a form.	Supplemental Lesson 11
6.4.4. Explain the concept of a form action.	Supplemental Lesson 11
6.4.5. Write the HTML code to add a working button (e.g., submit, reset) to a form.	Supplemental Lesson 11
6.4.6. Format a completed form using HTML and Cascading Style Sheets (CSS) (e.g., fieldset, tabindex).	Supplemental Lesson 11
6.4.6. Code scripting to interact with data sources (e.g., database, web services).	N/A
6.5. Websites: Create and update a website.	**CITATION(S)**
6.5.1. Implement web programming standards and protocols (e.g., World Wide Web Consortium [W3C], Hypertext Markup Language [HTML] 5).	Throughout the course.
6.5.2. Plan a website's structure for navigation and usability.	Chapter 13, Lesson 1
6.5.3. Utilize standard web programming languages (e.g., markup, scripting languages) in website development.	Throughout the course.
6.5.4. Install and configure a content management system (CMS).	N/A
6.5.5. Select an integrated development environment (IDE).	Chapter 16, Lesson 4
6.5.6. Create and edit a webpage template.	Chapter 11, Lesson 2 Chapter 16, Lesson 5 Chapter 24, Lesson 2 Chapter 25, Lesson 4

6.5.7. Create and attach cascading style sheets (CSS).	Chapter 6, Lesson 5, and throughout the course.
6.5.8. Format website layout (e.g., targeted platforms, text formatting, background color, text, tables, lists, iframes).	Chapters 7, 8, 10, 12, Supplemental Lesson 12
6.5.9. Incorporate audio and video, forms, and links on a website.	Chapters 5, 19 Supplemental Lesson 11
6.5.10. Develop and execute usability tests on a completed website, checking for information accessibility, ease of use, and navigation.	Chapter 13, Lesson 2 Chapter 14 Chapter 28
6.5.11. Code a website for cross-platform and cross-browser compatibility and validation.	Chapter 20, Lesson 1 Chapter 21, Lesson 3
6.5.12. Publish the completed website to a web server.	N/A