

SAN DIEGO COMMUNITY COLLEGE DISTRICT  
CONTINUING EDUCATION  
COURSE OUTLINE

**SECTION I**

**SUBJECT AREA AND COURSE NUMBER**

COMP 644

**COURSE TITLE**

SERVER SIDE WEB DEVELOPMENT

**TYPE COURSE**

NON-FEE

VOCATIONAL

**CATALOG COURSE DESCRIPTION**

This course includes the installation, configuration and management of a web server application. Students will learn about the structure and components of static and dynamic websites, programming web applications, and database management. Students will be introduced to web server software, programming languages used to create web applications, and how to use a database to create dynamic websites. (FT)

**LECTURE/LABORATORY HOURS**

170

**ADVISORY**

Microcomputer Basics or equivalent.

**RECOMMENDED SKILL LEVEL**

Possess a 10<sup>th</sup> grade reading level; ability to communicate effectively in the English language; knowledge of math concepts at the 8<sup>th</sup> grade level and basic computer literacy.

**INSTITUTIONAL STUDENT LEARNING OUTCOMES**

1. Social Responsibility  
SDCE students demonstrate interpersonal skills by learning and working cooperatively in a diverse environment.
2. Effective Communication  
SDCE students demonstrate effective communication skills.
3. Critical Thinking  
SDCE students critically process information, make decisions, and solve problems independently or cooperatively.

INSTITUTIONAL STUDENT LEARNING OUTCOMES (CONTINUED)

4. Personal and Professional Development  
SDCE students pursue short term and life-long learning goals, mastering necessary skills and using resource management and self-advocacy skills to cope with changing situations in their lives.

COURSE GOALS

1. Introduce the architecture of a client and web server.
2. Learn how to create a static web site.
3. Learn how to style a website.
4. Learn the use of administrative tools to manage a web server.
5. Learn how to code and test a web application.
6. Learn how to incorporate a web application in a dynamic web site.
7. Understand the purpose and structure of a database when used for a dynamic website.
8. Learn how to create and modify a database.
9. Learn how to use a web application to access and modify data in a database.
10. Learn how to display data from a database in a dynamic website.
11. Understand the purpose and appropriate application of the Model View Controller (MVC) pattern.
12. Understand and employ appropriate troubleshooting methodology.

COURSE OBJECTIVES

Upon successful completion of this course, students will be able to:

1. Describe the function and components of a web server and client.
2. Create a static website.
3. Employ style sheets on a website.
4. Demonstrate appropriate use of web server and database management tools.
5. Deploy a web application in a dynamic website.
6. Demonstrate how to run a web application.
7. Code a web application.
8. Describe the structure of a database.
9. Demonstrate how to use, create and modify databases.
10. Create statements to select, insert, update, and delete data in a database.
11. Demonstrate how to use code to get data from a database.
12. Devise code to modify and retrieve data in a database.
13. Devise code to display data on a web page.
14. Use the MVC (Model View Controller) pattern to develop web applications.
15. Test and debug web applications.
16. Deploy integrated web application.

**SECTION II**

**COURSE CONTENT AND SCOPE**

1. Client-Server Architecture
  - 1.1. Components of client-server architecture
  - 1.2. HTTP request and responses
  - 1.3. Static vs dynamic web page processing
  - 1.4. Software components of web applications
    - 1.4.1. Web server software
    - 1.4.2. JavaScript-based framework
    - 1.4.3. Server-side scripting
    - 1.4.4. Web browser
    - 1.4.5. Database
    - 1.4.6. AJAX technologies
    - 1.4.7. Event-based server execution procedures
  - 1.5. Web application deployment
    - 1.5.1. Local development environment
    - 1.5.2. Production internet server
  - 1.6. Components of an Hypertext Transport Protocol (HTTP) Uniform Resource Locator (URL)
  - 1.7. Integrated Development Environment (IDE)
    - 1.7.1. Benefits of an IDE
    - 1.7.2. Components of an IDE
    - 1.7.3. Using an IDE to create a website and application
2. Web Page Structure
  - 2.1. HTML syntax and introduction to Document Object Model (DOM)
    - 2.1.1. Elements
    - 2.1.2. Attributes
    - 2.1.3. Values
  - 2.2. Head elements
  - 2.3. Meta elements
  - 2.4. Body elements
  - 2.5. Structural elements
  - 2.6. Special Entity Characters
  - 2.7. Hyperlinks
  - 2.8. Page breaks
  - 2.9. Directory structure and scalability
3. Cascading Style Sheet (CSS)
  - 3.1. Purpose and advantages
  - 3.2. Types of style sheets
  - 3.3. CSS syntax
    - 3.3.1. Selectors
    - 3.3.2. Properties
    - 3.3.3. Attributes
    - 3.3.4. Values
  - 3.4. Page Layout
  - 3.5. Location of CSS directives and how to reference

COURSE CONTENT AND SCOPE (CONTINUED)

4. Web Page Elements
  - 4.1. Common uses
  - 4.2. Form elements
  - 4.3. Form controls
  - 4.4. Parsing DOM programmatically
5. Databases
  - 5.1. Types of databases
  - 5.2. Database structure
    - 5.2.1. Entity relationship diagrams
    - 5.2.2. Tuples
  - 5.3. Relational databases
    - 5.3.1. Primary and foreign keys
    - 5.3.2. Indexes
  - 5.4. Types of statements
    - 5.4.1. Select
    - 5.4.2. Insert
    - 5.4.3. Update
    - 5.4.4. Delete
    - 5.4.5. Union
    - 5.4.6. Joins
    - 5.4.7. Alter
    - 5.4.8. Drop
  - 5.5. Users and privileges
  - 5.6. Connections from web applications
    - 5.6.1. Prepared statements
    - 5.6.2. Sanitizing data
6. Server Side Scripting Language
  - 6.1. Installation, configuration, updates, extensions
  - 6.2. Code syntax
    - 6.2.1. Placement in document
    - 6.2.2. Tags
    - 6.2.3. Statements
    - 6.2.4. Comments
    - 6.2.5. Case sensitivity
  - 6.3. Variables and constants
    - 6.3.1. Declaration
    - 6.3.2. Naming rules
    - 6.3.3. Output variables
    - 6.3.4. Scopes
  - 6.4. Variable types
    - 6.4.1. String
    - 6.4.2. Integer
    - 6.4.3. Float
    - 6.4.4. Boolean
    - 6.4.5. Array
    - 6.4.6. Object
    - 6.4.7. Null

### COURSE CONTENT AND SCOPE (CONTINUED)

- 6.5. Casting
- 6.6. Output statements
  - 6.6.1. Echo
  - 6.6.2. Print
- 6.7. Operators
- 6.8. Control statements
- 6.9. Code to query database and generate HTML
  - 6.9.1. Basic login
  - 6.9.2. HTML table display from database table
- 6.10. Code to accept form data and insert into database
  - 6.10.1. Basic registration page
  - 6.10.2. Live data modification
- 6.11. Code to provide responsive user feedback
  - 6.11.1. Persistent connections
  - 6.11.2. Live data manipulation

### APPROPRIATE READINGS

Appropriate readings may include, but are not limited to, periodicals, magazines, instructor-written materials, manuals, instructor selected URLs, and publications related to the implementation of server operating systems and related web applications.

### WRITING ASSIGNMENTS

Appropriate writing assignments may include, but are not limited to, preparing text for an assigned project, documenting all laboratories and project work, and completing all written assigned reports.

### OUTSIDE ASSIGNMENTS

Outside assignments may include, but are not limited to, reading texts and reference resources; research as needed to complete projects; and organizing and preparing written answers to assigned questions.

### APPROPRIATE ASSIGNMENTS THAT DEMONSTRATE CRITICAL THINKING

Assignments which demonstrate critical thinking may include, but are not limited to, analysis and evaluation of assigned text and reference resources, and utilize this analysis in classroom discussions, writing assignments, and in performing laboratory activities. Students must select appropriate methods and resources needed to complete laboratory assignments.

### EVALUATION

A student's grade will be based on multiple measures of performance and will include evaluation of student's ability to:

1. Perform in a variety of activities and assignments related to the course objectives.
2. Complete written and practical examinations.

EVALUATION (CONTINUED)

3. Contribute to class and group discussions.
4. Maintain attendance and punctuality per current policy.
5. Demonstrate ability to work independently and as a team member.
6. Demonstrate troubleshooting skills.

Upon successful completion of each course in the program, a Certificate of Course Completion will be issued. Upon successful completion of all courses included in the program, a Certificate of Program Completion will be issued.

METHOD OF INSTRUCTION

Methods of instruction may include, but are not limited to, lectures, self-paced lab, demonstrations, individualized study, use of audio-visual aids, group/team work, tutorials, outside assignments, guest lectures, field trips, and guided student job assignments. This course, or sections of this course, may be offered through distance education.

TEXTS AND SUPPLIES

*Basics of Web Design: HTML5 & CSS3*, Terry Felke-Morris, Pearson, current edition  
*Murach's PHP and MySQL*, Joel Murach and Ray Harris, Mike Murach & Associates, current edition  
*Learning PHP, MySQL & JavaScript*, Robin Nixon, O'Reilly, current edition  
*Node.js the Right Way: Practical, Server-Side JavaScript That Scales*, Jim R. Wilson, Pragmatic Programmers, current edition

Web Resources: <http://www.w3schools.com>; <http://www.apachefriends.org>; <http://php.net>;  
Supplies: Journal (composition book), USB Drive or other storage media

PREPARED BY: Richard Gholson and Duane Rinehart      DATE: 10/13/15

REVISED BY: \_\_\_\_\_      DATE: \_\_\_\_\_

Instructors must meet all requirements stated in Policy 3100 (Student Rights, Responsibilities and Administrative Due Process), and the Attendance Policy set forth in the Continuing Education Catalog.

REFERENCES:

San Diego Community College District Policy 3100  
California Community Colleges, Title 5, Section 55002  
Continuing Education Catalog