



Clemson University - Center for Corporate Learning
1 North Main Street, 7th Floor,
Greenville, SC 29601
<http://www.clemson.edu/online/>
Contact: Juanita Durham | 864.656.3984 | jdrhm@clemson.edu

Full Stack Web Developer with MEAN Stack

Format: Self-Pace Online / eLearning
Program Duration: 6 Months
Course Contact Hours: 375
0

Full Stack Developer with MEAN Profession

MEAN Stack Developer is a software engineer who is specialized in developing web applications with so called MEAN, a collection of JavaScript-based technologies. Responsibilities include building interactive consumer data from multiple systems and RESTfully abstract to the UI through a Node.js backend; define code architecture decisions to support a high-performance and scalable product with a minimal footprint. Address and improve any technical issues; and collaborate well with engineers, researchers, and data implementation specialists to design and create advanced, elegant and efficient systems.

Full Stack Developer with MEAN Program

The MEAN Stack Web Developer Complete Video Course introduces students to building dynamic web applications with MongoDB, Express, AngularJS, and Node.js. Students learn about essential JavaScript-based technologies, async programming, and MVC design techniques. Students also walk through establishing database connectivity and secure user authentication. Once complete, students will have the basic skills they need to create full-fledged JavaScript-based web solutions from server to browser, and integrate all MEAN stack technologies into full working systems.

Education and National Certifications

- Students should have or be pursuing a high school diploma or GED.
- Students who complete this program can sit for the **Microsoft Technology Associate (MTA) 98-375: HTML5 Application Exam**

Program Objectives

At the conclusion of this program, students will be able to:

- Manage the application development life cycle
- Build the user interface using HTML5
- Format the user interface using CSS
- Code the user interface using JavaScript
- Install and configure all components of the MEAN stack
- Work with the JavaScript features all MEAN stack developers need to know

- Write and extend JSON web servers and Node modules
- Simplify web development with Express web servers
- Build AngularJS apps with directives, filters, MVC, and modules
- Create databases, collections, and documents with MongoDB
- Authenticate your users
- Support microposts, followers, and other user collaboration features

Full Stack Web Developer with MEAN Stack Detailed Student Objectives

PRE-REQUISITE: HTML & CSS

INTRODUCING HTML

- Understand the basics of HTML
- Choose a tool for writing HTML
- Post HTML to a hosting service
- Define HTML5

BUILDING A BASIC HTML DOCUMENT

- Build a simple web page
- Recognize the standard HTML tags for every web page
- Add paragraphs and line breaks
- Create HTML headlines
- Use semantic tags
- Write links to other web pages

INTRODUCING CSS

- Define CSS
- Use inline styles
- Write a basic CSS style sheet
- Use internal and external style sheets
- Add CSS Classes and IDs
- Understand the Basics of CSS

BUILDING AND STYLING HTML TEST

- Align Text
- Write ordered and unordered lists
- Create definition lists
- Add two types of quotations

UNDERSTANDING HTML5 SEMANTIC ELEMENTS

- Recognize sectioning elements
- Add figures
- Include document headers and footers
- Create HTML5 outlines

ADDING IMAGES AND MULTIMEDIA

- Add images to web pages
- Position images
- Use background images
- Link to documents and PDFs
- Add HTML5 video and audio
- Make multimedia responsive

BUILDING AND USING HTML FORMS

- Create a basic form
- Use HTML5 input
- Add in other form input fields
- Validate forms and Submit form data

JAVASCRIPT

GETTING STARTED WITH JAVASCRIPT

- Understand how JavaScript is used
- Practice interactively with JavaScript (simple and complex versions)
- Define, modify and access variables
- Apply operators
- Define and call functions

JAVASCRIPT BASIC SYNTAX – PART 1

- Execute code conditionally
- Execute code
- Format code so that it can be understood, edited and maintained
- Call functions from the Math class
- Embed JavaScript in HTML
- Choose among HTML versions

JAVASCRIPT BASIC SYNTAX – PART 2

- Define and access arrays
- Create, compare, and call methods on strings
- Recognize patterns with regular expressions
- Call methods on arrays

JAVASCRIPT FUNCTIONS

- Understand how the use of functions in JavaScript differs from other languages
- Assign functions to variables, arrays, and other data structures
- Pass functions to other functions
- Return functions from functions
- Create anonymous functions
- Capture local variables (make closures)
- Use the apply method

JAVASCRIPT OBJECTS

- Define, instantiate, and access objects

- Define methods with the prototype property
- Use namespaces to avoid name conflicts
- Create anonymous objects
- Create and parse JSON
- Apply a few other object "tricks"

HTML & CSS

HTML BASICS

- Differentiate among HTML versions
- Define hypertext links and apply different types of URLs
- Understand how block-level (paragraph-like) elements differ from other HTML elements
- Use headings and basic paragraphs
- Understand the difference between div and span
- Create bulleted and numbered lists
- Understand how inline (font-line or text-level) elements differ from other HTML elements

MORE HTML

- Use tables to present tabular information or to align sections of text
- Use forms to send data to servers
- Collect user data with form input elements
- Apply a few miscellaneous elements

jQuery

JQUERY INSTALLATION, OVERVIEW AND GETTING STARTED

- See what jQuery is all about
- Download jQuery and use it in your pages
- Interactively practice with jQuery
- Manipulate the DOM with jQuery
- Walk through a quick example of the use of jQuery selectors

DEPLOYING WEB APPLICATIONS WITH ECLIPSE AND TOMCAT

- Understand why you need to run on a "real" web server when applying Ajax
- Install Java, Tomcat, and Eclipse
- Configure and launch Eclipse
- Create and deploy apps in Eclipse

JQUERY AJAX SUPPORT - BASICS

- Understand what Ajax is all about and how its use improves traditional web applications
- Make Ajax requests with \$.ajax: basics

- Walk through a complete example that uses \$.ajax
- Define jQuery behaviors with unobtrusive JavaScript
- Insert Ajax results into the page
- Implement content-centric Ajax with the load function

jQuery AJAX SUPPORT – SENDING DATA TO THE SERVER

- Send data to the server with \$.ajax: summary
- Send data with an explicit string
- Send data with a data object
- Send data by using the serialize function
- Send data when using the load function

jQuery AJAX SUPPORT ADVANCED CAPABILITIES

- Display temporary messages during Ajax requests
- Handle JSON data
- Walk through an example of handling JSON Data
- Understand and apply \$.ajax options
- Use shortcut functions

jQuery AJAX SUPPORT – PROMISES AND DEFERRED OBJECTS

- Using promises and deferred objects for Ajax
- Highlight Ajax results
- Display temporary messages during Ajax requests
- Check multiple promises with \$.when
- Create and apply your own deferred objects

jQuery: SELECTORS AND DOM MANIPULATION

- Review the basics of CSS
- Recall the basics of jQuery DOM manipulation
- Find sets of page elements with basic selectors
- Operate on sets of page elements with basic operators
- Respond to user events
- Walk through an example that combines selectors, operators and event handlers

UNIT TESTING WITH JsUNIT

- Understand the purpose of JsUnit and the basic steps in applying it
- Install JsUnit and begin using it
- Use available assertion functions
- Walk through a more realistic example
- Apply unit testing to functions that use jQuery

jQuery UI BASICS

RICH INTERFACES WITH jQuery UI SETUP AND BASIC WIDGETS

- Understand what jQuery is all about and how to set up your page to use it
- Use styled buttons: basics
- Use styled buttons: options
- Collect dates using a date picker (popup calendar)

- Collect numbers using a number spinner
- Collect numbers using a slider
- Show updates using a progress bar

jQuery UI Dialog Boxes

- Understand what dialog boxes are for
- Create and pop up basic dialogs
- Animate the opening and closing of dialogs
- Make dialogs modal
- Use modal dialogs that have been inserted automatically
- Display Ajax content in dialogs

jQuery UI Tabbed Panels and Accordion Panels

- Display static content in tabbed panels
- Display dynamic (Ajax) content in tabbed panels
- Animate the opening or closing of tabs
- Display content in accordion panels

jQuery UI Autocompleters

- See how the page is set up for the lesson examples
- Understand what autocompleters are for and the basic alternatives in the source of the suggestions
- Get suggestions from local (client-side) data
- Get suggestions from remote (server-side) data
- Get suggestions from a custom matching function
- Apply custom CSS styles to the suggestion menu
- Use an array of objects to differentiate between display values and insert values
- Perform side effects in addition to inserting value into textfield

jQuery UI Animation Effects and Themes

- Understand what animation effects are all about and what the available effects are
- Walk through an example that applies to each of the animation effects
- Understand how themes let you update the look of your pages with little change to the code
- Change themes at runtime
- Design your pages so that they adapt well when the theme changes

Using Git and GitHub in Web Development

Configuring Git

- Understand three levels of configuration
- Deploy basic configuration settings
- Configure line endings
- Configure aliases

Getting Started with GitHub

- Create your first Git repository
- Commit in Git
- Understand a Git commit

- Understand the benefits of the staging area
- Use the Git log to view history

FILES IN GIT – RENAMING, DELETING AND IGNORING

- Rename a file in Git
- Delete a file in Git
- Ignore files using a .gitignore file
- Use global gitexcludes and other Git ignore options
- Understand Git ignore precedence
- Use Git commit -a

FILES IN GIT – BRANCHING, MERGING AND REBASING

- Understand branching
- Merge a branch
- Create a fast forward merge
- Understand recursive merges
- Understand "no fast forward" recursive merges
- Resolve merge conflicts
- Understand a merge conflict example
- Understand Git Diff
- Understand rebasing
- Rebase a branch
- Handle rebase conflicts

REVIWEING A PROJECT ON GITHUB

- Get an overview of a project on GitHub using the README
- Get more information about a project
- Understand issues
- View project state through pulse and graphs

CONFIGURING A PROJECT ON GITHUB

- Understand basic configuration options and rename a repo
- Understand GitHub Pages
- Add collaborators and integrations
- Configure deploy keys

MEAN STACK WEB DEVELOPMENT

Learning Node.js

GETTING STARTED WITH NODE.js

- Install Node.js on Windows
- Install Node.js on the Mac
- Install Node.js on Linux
- Create a Node.js application
- Create a Node.js module
- Utilize the node.js callback model
- Implement events and listeners

- Handle streams and file I/O
- Run your first web server
- Use the Node.js debugger

A CLOSER LOOK AT JAVASCRIPT

- Use basic types
- Use strings
- Use objects
- Use arrays
- Make full use of functions
- Use language constructs
- Write classes with prototypes and inheritance
- Use handle errors
- Work with two important Node.js globals

ASYNCHRONOUS PROGRAMMING

- Compare synchronous and asynchronous programming
- Mix error handling and synchronous programming
- Solve a new problem that arises in Node.js-losing your “this” reference
- Yield control and improve responsiveness
- Take a quick look at synchronous

WRITING APPLICATIONS

- Write your first JSON server
- Combine loops and asynchronous programming
- Support multiple request types in your server
- Explore requests and responses
- Add support for query (GET) parameters
- Receive POST data

MODULES

- Writing a simple module
- Use npm-the node package manager
- Write a more interesting module
- Publish modules
- Explore the async module – a vital component of our apps

EXPANDING OUR WEB SERVER

- Investigating Node.js streams
- Serve static files from our server
- Serve different file types with our server

BUILDING WEB APPLICATIONS EASIER WITH EXPRESS

- Install express and write Hello World
- Route requests
- Update your app for routing
- Understand REST API design
- Factorize your code with modules

- Explore connect middleware
- Clarify the importance of ordering in middleware
- Replace static file handling in the app with middleware
- Support POST data cookies and sessions
- Upload files
- Implement HTTP basic authentication
- Implement Virtual Hosting with Express

LEARNING ANGULAR JS

GETTING STARTED

- Understand what AngularJS is and why it's important
- Install AngularJS
- Install Bootstrap
- Write your first AngularJS application
- Look at what you will build with AngularJS
- Understood what you should know before you begin

WORKING WITH ANGULARJS BUILDING BLOCKS: BINDING-DIRECTIVES AND FILTERS

- Perform more useful data binding with repeating
- Introduce filters to your bindings
- Combine data binding and filters in your application
- Understand more about repeating and directives
- Improve the UI of your sample application

USING CONTROLLERS AND VIEWS

- Understand how AngularJS organizes MVC
- Use controllers and forms to modify your data
- Perform data validation and communicate between views and controllers
- Understand validation a bit more
- Set up a simple web server (Mac and Linux)
- Set up a simple web server (Windows)
- Set up multiple views and controllers

PUTTING IT ALL TOGETHER WITH MODULES

- Understand the core components of AngularJS modules
- Understand routes and add them to your apps
- Understand how to lay out your applications
- Add an album view to your sample application

LOADING AND SAVING DATA

- Understand the various ways you can provide data in Angular-Part 1: Services
- Understand the various ways you can provide data in Angular-Part 2: Services
- Use factors and services with remote data
- Upload image files from your AngularJS application

FILLING IN THE BLANKS-FILTERS, DIRECTIVES, COOKIES, AND MORE

- Write your own filters
- Write your own directives I: simple directives
- Write your own directives II: scope and user interface
- Work with cookies in your app

USING MONGODB WITH NODEJS AND ANGULARJS

MONGODB FUNDAMENTALS

- Create databases, collections, and documents in the MongoDB Shell
- View documents in a collection from the MongoDB Shell
- Modify and remove documents collections and databases from the MongoDB Shell
- Apply users and authentication to MongoDB
- Add and remove indexes to improve performances

USING MONGODB WITH NODEJS

- Set up and run Mongo DB
- Perform basic Mongo DB operations
- Query collections with the “find” method
- Update your photo app – setting everything up
- Add new database operations to the app
- Add new pages and templates to the app

IMPLEMENTING FULL NODESJS, MONGODB & ANGULARJS STACK TO ADD USER AUTHENTICATION TO WEBSITE

- Create the application server routes and views
- Add session support to the server
- Implement the user controller to interact with MongoDB
- Implement the AngularJS user model and controller