**Full Stack Web Development Certificate**

The program provides students the skills required to work as web developer building websites for large or small businesses. Students will gain the knowledge necessary to build front-end website designs, back-end database driven web systems, and be able to use many of the common tools used today in the industry, such as HTML, CSS, JavaScript, PHP, MySQL, and WordPress.

Below are the course descriptions for each of the modules within the Full Stack Web Development Non-Credit Certificate.

|  |  |  |
| --- | --- | --- |
| Module | Name | Hours |
| 1 | Basics of Computer Usage and Web Development | 8 |
| 2 | Hyper-text Markup Language (HTML) | 12 |
| 3 | Cascading Stylesheets (CSS) | 16 |
| 4 | Debugging and Development Tools | 8 |
| 5 | ASSESSMENT 1 – Basic HTML Scripting and Debugging | 2 |
| 6 | JavaScript | 24 |
| 7 | Search Engine and Social Media Optimization | 4 |
| 8 | Working with APIs | 12 |
| 9 | ASSESSMENT 2 – Basics of Front-End Development | 4 |
| 10 | Networking, Protocols, Web Hosting and Security | 8 |
| 11 | Server-Side Programming - PHP | 24 |
| 12 | Databases and SQL – MySQL | 12 |
| 13 | ASSESSMENT 3 – Basics of Back End Development | 4 |
| 14 | MTA Certification Prep | 10 (150 shared) |
| 15f | Advanced JavaScript | 24 |
| 16f | ES6 | 24 |
| 17f | ReactJS | 24 |
| 18f | Advanced Concepts (CORS, Package Management, Storage) | 24 |
| 19f | ASSESSMENT 4 – Intermediate Front-End Development | 4 |
| 15b | Server-Side Programming – TBD (Java, Go, Python) | 24 |
| 16b | Basic Algorithms and Data Structures | 24 |
| 17b | Advanced MySQL and NOSQL Databases | 24 |
| 18b | REST / SOAP API Development | 24 |
| 19b | ASSESSMENT 4 – Intermediate Back-End Development | 4 |
|  |  |  |
| Total |  | 250 |

**MODULE:** Basics of Computer Usage and Web Development

**COURSE DESCRIPTION:**

This eight (8) hour course provides the knowledge in basic computer usage and the tools necessary to being web development. We introduce the idea of an Integrated Development Environment, HTTP, Web Browsers, and basic tools necessary for web development. We also introduce tool necessary to understand how to move around your computer, like the command line and modifying files to change the behavior of your computer. We will also focus on the importance of documentation and the requirement that all code written should be documented for it to be considered complete.

**DURATION:** 8 Hours (0 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module, the trainee will be able to:

* Understand the difference between a local computer and the Internet, identify the parts of a computer and the basics of computer networking and how the Internet works.
* Understand the basics of using command line tools to move around on a computer
* Understand what and Integrated Development Environment is and how to use it to be able to design websites.
* Understand the difference between types of web browsers and how to use basic debugging tools available to you in the browser to view different types of web pages.
* Understand HTTP(S), IP Addresses, Localhost, and the difference between a development environment and a live website
* Understand the basics of evolution of web development and the difference between front-end and back-end development

**MODULE:** Hyper-text Markup Language (HTML)

**COURSE DESCRIPTION:**

This twelve (12) hour course provides foundational skills necessary to build a basic web page using HTML. We will cover tags, attributes, values, and how to nest and embed those tags in order to build a complex web layout. We will also review the basics of wireframes and how to lay out web pages based on wireframes. Also, we will cover new, and important concepts, such as WCAG II and accessibility (A11Y), modern naming structure conventions (BEM), and search engine optimization (SEO)

**DURATION:** 12 Hours (8 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module, the trainee will be able to:

* Be able to take a basic wireframe and build a layout in HTML
* Understand what an HTML tag, attribute, and value is
* Understand how to build complex layouts using only HTML
* Understand the limitations of HTML and why we need more complex systems to build most modern webpages
* Understand the basics of images, different types and how to use them in HTML files
* Understand the difference between HEAD and BODY
* Understand basic types of HTML divider tags and why special versions exist
* Craft accessible websites for all types of users
* Create architectures that are SEO friendly and work fluidly with JavaScript.

**MODULE:** Cascading Stylesheets (CSS)

**COURSE DESCRIPTION:**

This sixteen (16) hour course provides the necessary knowledge to be able to use Cascading Stylesheets in web pages to be able to fine tune the control and design of web pages to make them appear more modern. Will cover inline, internal, and external stylesheets. Will also review ID and Class attributes and how each work separately. Will also cover CSS specificity and building, or troubleshooting, on the fly through browser tools.

**DURATION:** 16 Hours (8 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module, the trainee will be able to:

* Understand how to build complex designs using Cascading stylesheets
* Understand the difference between IDs and Classes
* Define CSS selectors and how and why you need to be able to target different tags separately
* Be able to calculate the CSS specificity of a given CSS attribute
* Understand and implement most CSS attributes and be able to explain the differences and functions of most of them.
* Should be able to implement the design of a complex HTML layout with CSS, like nytimes.com or similar websites
* Learn the BEM architecture
* Have a basic understanding of different frameworks, particularly Material UI
* Understand and fully utilize the full spectrum of layout methods (Grid, Flex, Float, etc.)
* Understand responsive design and its importance
* Understand CSS pre-processors such as SCSS, Less, and Postcss.

**MODULE:** Debugging and Development Tools

**COURSE DESCRIPTION:**

This eight (8) hour course provides the necessary knowledge to be able to utilize source control tools like GitHub and the Integrated Development Environment (IDE) to debug and troubleshoot issues. The focus on this module puts emphasis on Git and source control.

**DURATION:** 8 Hours (4 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module, the trainee will be able to:

* Understand how to initialize a git repository
* Understand how remote repositories work and how to push and pull from a remote
* Understand how to fix merge conflicts and how to use the IDE to help locate and resolve them.
* Be able to work as a team to make commits on the same repository without creating merge conflicts
* Understand git workflow and proper branching methods
* Understand media queries and how they are used to build “responsive” web templates.
* Understand @print directive and how to format web pages for printing.

**ASSESSMENT:** Upon completion of this module there will be a 2-hour assessment test to gauge understanding of the core competencies. This assessment will be a 1 hour written exam and a 1-hour project to be done in class.

# MODULE: JavaScript

**COURSE DESCRIPTION:**

This twenty-four (24) hour course is designed to provide a complete understanding of “vanilla” JavaScript including control structures, data types, variables, functions, classes, and object-oriented design techniques. We will cover core and theoretical concepts, such as hoisting, scope, and the prototype chain.

**DURATION:** 24 Hours (8 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Build simple JavaScript applications, like a calculator that can perform basic math and data manipulation functions.
* Understand the difference between string, int, float, and array data types
* Understand the 4 basic control structures that exist in JavaScript
* Understand the 3 major concepts of object-oriented programming and which exist in JavaScript
* Understand JavaScript Object Notation and how to construct a basic object
* Be able to build basic functions and classes and reuse those functions in other files.
* Understand the difference between embedded and external JavaScript files
* Understand minification of JavaScript (and CSS) and why it is beneficial.
* Understand how to integrate JavaScript into HTML and CSS and to have JavaScript manipulate both languages dynamically.
* Understand the basics of JavaScript and its security limitations
* Understand concepts of hoisting, scope, and prototype chain

**MODULE:** Search Engine and Social Media Optimization

**COURSE DESCRIPTION:**

This four (4) hour course explains how Search Engine Optimization is done in page and explains best practices for building SEO optimized websites and writing clean code that search engines understand. Additionally, this class covers social media optimization and how to make your web pages look professional when shared on various social media platforms.

**DURATION:** 4 Hours (0 Lab Hours)

**LEARNING OBJECTIVES:** Upon completion of this module, the trainee will be able to:

* Identify various search engine platforms and their market share
* Understand how various social media platforms are used by websites and apply different types of features to each.
* Understand how to use Google and Bing Webmaster tools to check the status of SEO efforts.
* Understand the correct meta tags and location within an HTML document for each search engine.
* Understand the OpenGraph and how it is used by various platforms like Twitter and Facebook.

**MODULE:** Working with APIs

**COURSE DESCRIPTION:**

This twelve (12) hour course introduces the concept of an Application Programming Interface (API) and how web sites work with APIs to build dynamic real-time content. JSON and XML API responses will be discussed along with the basics of RESTful API methods.

**DURATION:** 12 Hours (4 Lab Hours)

**LEARNING OBJECTIVES:** Upon completion of this module, the trainee will be able to:

* Identify what an API is
* Explain how to use JavaScript to interface with an API and to load content dynamically into a web page.
* Understand the 4 main methods used in REST APIs and their purposes
* Understand how to use Postman to send test API requests and to test responses
* Understand how JSON web tokens and basic REST API security works across various APIs
* Be able to use many of the following APIs, Google Maps, Twitter, Facebook, Weather Underground

**ASSESSMENT:** Upon completion of this module there will be a 4-hour assessment test to gauge understanding of the core competencies of front-end web development. This assessment will be a 1 hour written exam and a 3-hour project to be done in class. There will also be a take-home component of this exam that the trainee should complete and upload to GitHub for final review.

**MODULE:** Networking, Protocols, Web Hosting and Security

**COURSE DESCRIPTION:**

This eight (8) hour course introduces some core concepts necessary for back-end web development. The trainee will be introduced to basic networking concepts like IP addressing, subnetting, HTTP and FTP protocols, and basic application security. It also covers the basics for hosting a website and managing and maintaining the servers that host it. In this course, we introduce shared hosting, dedicated hosting, DNS, and basic maintenance of servers and software. We will build a basic WordPress web hosting server, along with build a web host from scratch using Linux, Apache, MySQL and PHP (LAMP stack). We will review security best practices including firewall basics and security of access to systems. We will also review various web hosting environments like GoDaddy, AWS, Azure, and Rackspace. We will outline some of the security issues with front-end web development and explain why backend development is necessary.

**DURATION:** 8 Hours (4 Lab Hours)

* Define IP address, subnetting, DHCP, DNS, and protocols
* Be able to name the more common protocols in use by web applications today.
* Explain the difference between a standard connection and a web socket
* Understand the security issues with passing passwords and other sensitive information in plain text.
* Understand the responsibility of a programmer when it comes to application security
* Understand how to setup a web server to host a website and serve HTTP pages
* Understand basic web hacks, like SQL injection and variable and browser manipulation
* Understand how to host a website that is available on the Internet for anyone to view
* Setup a domain name and point it to a web server
* Configure apache web servers
* Understand basic linux commands and how to keep the server up to date and protected against security threats
* Have a basic understanding of Amazon Web Servers, GoDaddy, Microsoft Azure and Rackspace
* Understand proper server architecture
* Identify security holes in web servers and be able to correct them
* Understand the basics of firewall security and limiting access to servers in various configurations

**MODULE:** Server-Side Programming (PHP)

**COURSE DESCRIPTION:**

This twenty-four (24) hour course introduces server-side programming to the trainees with a focus on PHP programming. We will focus on the differences between front-end and back-end programming and why you would choose each option. We will introduce the concept of fully object-oriented programming and building massively scalable applications using PHP. We will also introduce the concepts of connecting to a database and the MVC framework for developing applications in PHP. We will also introduce a second server-side programming language, Python, and explain the differences between the two.

**DURATION:** 24 Hours (8 Lab Hours)

* Understand the syntax and usage of PHP within an application
* Understand includes, arrays, array functions and the require and require\_once functions
* Understand the built-in functions that are available to use with PHP and basic third party addons
* Be able to identify and build a basic WordPress website and make modifications to the theme and plugin files
* Be able to build a simple HTTP application in Python which acts as a RESTful API.
* Understand how to connect to a simple database and run commands against that database to pull data.

**MODULE:** Databases and SQL – MySQL

**COURSE DESCRIPTION:**

This twenty-four (24) hour course teaches the basics of the SQL language. The focus is on SELECT, INSERT, UPDATE, DELETE statements, as well as JOINS, ORDER BY, and GROUP BY clauses. The trainee should be able to pull any set of complex data from a large database. Students will also learn best practices for indexing of tables and speeding up complicated queries. Basic MySQL administration will also be taught and students will know how to use MYSQL Workbench, along with PHPMyAdmin to import and export data.

**DURATION:** 22 Hours (8 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module, the trainee will be able to:

* Define database, table, column, row, query and different data types
* Be able to identify the correct data type for various types of input
* Understand the different types of queries available in the SQL language
* Be able to filter queries using the WHERE clause
* Be able to join multiple tables together to link different sets of data by a common parameter
* Understand database normalization and how to prevent data duplication
* Be able to sort, filter, and group data in a database.
* Be able to export data from excel and import it into a mysql database
* Understand how to use MySQL Workbench and PHPMyAdmin to administer a database
* Be able to use the CREATE table syntax to build a table with both primary and secondary indices.
* Understand the types of indexes available and identify which fields in a database should be indexed for optimal read/write throughput.

**ASSESSMENT:** Upon completion of this module there will be a 4-hour assessment test to gauge understanding of the core competencies of backend development. This assessment will be a 1-hour written exam and a 3-hour project to be done in class.

**MODULE:** MTA Certification Preparation

**COURSE DESCRIPTION:**

This ten (10) hour course is focused on preparing students for the MTA certification. Students will review the core concepts necessary to pass both the MTA certification for HTML/CSS and the MTA certification exam for Javascript. These exams cover the concepts that we have covered up to this point, but we will review the major concepts to ensure students are ready for the exams.

<https://www.microsoft.com/en-us/learning/exam-98-383.aspx>

<https://www.microsoft.com/en-us/learning/exam-98-382.aspx>

**DURATION:** 10 Hours (0 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Take and pass the MTA certification requirements for HTML/CSS and JavaScript fundamentals.

# MODULE: Advanced JavaScript

**COURSE DESCRIPTION:**

This deep dive into JavaScript is designed to provide a complete understanding of “vanilla” JavaScript including declarative DOM manipulation with such constructs as mutation observer, intersection observer, and more. We will cover core and theoretical concepts, such as hoisting, scope, and the prototype chain.

**DURATION:** 24 Hours (6 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Learn how the manipulation observer works
* Learn how the intersection observer works
* Learn about closures
* Learn hoisting and scope
* Learn event bubbling
* Learn about immediately invoked function expressions (IIFEs)

# MODULE: ES6 JavaScript

**COURSE DESCRIPTION:**

This module will teach the students how to use best features of modern JavaScript such as classes, arrow functions, destructuring, block scope, variable declaration, imports, etc. They will learn the advantages of compiled JS. They will learn about object oriented and functional JavaScript.

**DURATION:** 24 Hours (6 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Learn the new features of JavaScript
* Learn Object Oriented JavaScript
* Learn about template literals how to perform functions within strings
* Understand ES6 syntactic sugar and classes
* Full understanding of build tools such as Webpack

# MODULE: React.js

**COURSE DESCRIPTION:**

This module is about the increasingly popular React.js framework built on ES6 JS. Students will learn the power of React.js and how to create progressive web apps. They will learn the basic building blocks of React, such as state and props. They will understand the ease of using components. Learn about the context and suspense APIs

**DURATION:** 24 Hours (6 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Create a login progressive web app that interfaces with an API to authenticate a user and channel specific access
* Learn JSX, the main templating language of React
* Understand the security limitations of React applications
* Understand how to build different runtime environments (dev / live) with a React application
* Understand how to write, build, test, and deploy a React application
* Understand the difference between state and props inside a React application
* Be able to build React components that can be used and reused in various applications
* Build a basic login application that interfaces with an API to authenticate a user and provide them access to various services.
* Understand the security limitations of React applications
* Understand progressive web apps (PWAs), declarative programming, and React.js
* Comprehensive knowledge of package managers such as NPM

# MODULE: Advanced Concepts

**COURSE DESCRIPTION:**

Here students will learn advanced concepts such as node.js, npm package management, CORS, and interacting with browser storage: cookies, local storage, and session storage. Concepts of performance will also be studied.

**DURATION:** 24 Hours (6 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Learn the file system module
* Learn the HTTP module
* Learn about streams and handling files
* Take a csv file and learn how to create a graph with the data
* Learn how to upgrade, repair, and fully utilize third party packages.
* Learn about the performance of staged compilation
* Learn function inlining and garbage collection

**MODULE:** Server-Side Programming (Python, Java, or Golang)

**COURSE DESCRIPTION:**

This twenty-four (24) hour course introduces server-side programming to the trainees with a focus on a new, more advanced programming language. We will focus on more advanced topics such as design patterns, packages, and basic data structures.

**DURATION:** 24 Hours (12 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Understand the syntax and usage of PROGRAMMING LANGUAGE within an application
* Understand how to use packages and survey various packages available
* Be able to read CSV and TXT files in a programming language and then manipulate that data.
* Understand the basic concepts of distributed programming and how to build scalable web applications
* Build a full web application that uses best practices in programming design, security, scalability and execution speed.

**MODULE:** Basic Algorithms and Data Structures

**COURSE DESCRIPTION:**

This twenty-four (24) hour course introduces the concepts of algorithms and data structures to the developer. We will cover basic divide and conquer algorithms and recursion and introduce other algorithmic concepts at a high level. We will also cover the basics of common data structures like hashmaps, trees, lists and dictionaries.

**DURATION:** 24 Hours (12 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Understand how to write a recursive algorithm and some common problems where recursion can help improve the speed of an algorithm.
* Understand divide and conquer algorithms and how to use them for tasks like sorting
* Be able to identify other algorithmic concepts like dynamic programming and greedy programming at a high level.
* Understand list, hashmap, and dictionary data structures and various use cases for when they can be beneficial in programming
* Understand tree data structures and how to do a basic tree traversal

**MODULE:** Advanced MYSQL and NOSQL databases

**COURSE DESCRIPTION:**

This twenty-four (24) hour course introduces advanced concepts in database design. We will review features of MYSQL databases, such as grouping, and SQL operations. We will cover normalization of databases and learn how to optimize slow queries. Additionally, we will introduce MongoDB as a NOSQL database and explain the differences of when to use NOSQL compared of MYSQL databases.

**DURATION:** 24 Hours (12 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Understand various MYSQL operations such as COUNT, GROUPBY, DISTINCT, DATE and other advanced SQL commands.
* Understand how to use DESCRIBE on a SQL query to analyze potential slow queries.
* Understand how database INDEXES work and when you should create a new index.
* Be able to convert a database into normal form and be able to ALTER database tables to adapt to changing requirements
* Understand the difference between NOSQL and relational databases and the pros and cons of both.
* Be able to build a MONGODB database and then run basic commands to query data from it.

**MODULE:** REST / SOAP API Development

**COURSE DESCRIPTION:**

This twenty-four (24) hour course is focused on developing a RESTful API in which anyone, with the API documentation can interface. Students will work in groups and will start by building a series of routes and then having those routes talk to a database. Responses from the REST API will be in JSON. Proper documentation techniques are required. The API will be written in Python or Golang. Once completed then another student group will use the API to write a front-end application that interfaces with the API directly using only the documentation provided.

**DURATION:** 24 Hours (16 Lab Hours)

**LEARNING OBJECTIVES:**

Upon completion of this module the trainee will be able to:

* Build a RESTful API in Python or Golang which uses best practices for returning data and interfaces with a database.
* Build a front-end application in PHP or JavaScript that interfaces with the RESTful API written by another student
* Be able to read / write API documentation using best practices learned.

**ASSESSMENT:** Upon completion of this module there will be a 4-hour assessment test to gauge understanding of the core competencies of the course. This assessment will be a 1-hour written exam and each student will need to provide a 15-minute presentation on either a project that they developed in this course, or a topic that they have researched on their own related to the class.