Full-Stack .NET Development (Accelerated)

COURSE SYLLABUS

Locations

TRIAD CAMPUS
1231 Shields Rd. Suite 5
Kernersville, NC 27284
336.231.8632

Instructors

**Antonio Raynor*, Director of Education
Instructor/Mentor***
araynor@coderfoundry.com

Visiting Instructors

Jason Twichell, Instructor/Mentor
Mark Jang, Instructor/Mentor

Prerequisites for Admission

1) High school graduation or equivalency, as certified by any of the following:

   a. Proof of graduation from a public or private high school that operates in compliance with State or local law;

   b. Proof of graduation from a state registered home high school;

   c. Certificate of high school equivalency (GED);

   d. If the student is unable to obtain a copy of the student's high school transcript or certificate of high school equivalency, the student must provide Coder Foundry with written evidence of said inability and furnish a signed, notarized attestation of either graduation from a public or private high school that operates in compliance with State or local law, graduation from a state registered home school, or receipt of a certificate of high school equivalency;

   e. For persons at least 18 years of age who did not graduate from a public, private, or state registered home high school or obtain a certificate of high school equivalency, demonstration of an ability to benefit as determined by any test instrument approved by the North Carolina State Department of Education.

2) Personal interview with Coder Foundry staff.

3) Successful completion of a series of online tutorials designed to prepare students for success in the course.

4) Completion and submission of JavaScript coding exercises.

Required at start of class:

1) Personal laptop meeting the following requirements:
   - 2.0 GHz or faster processor
   - 8 GB RAM or better
   - HDMI output
   - Windows 7 or newer operating system (a MacBook running Windows in a virtual environment such as BootCamp™ or Parallels™ is acceptable, provided the above requirements are met).

2) Software installations (all free versions):
   - Visual Studio 2017 Community edition
     http://www.visualstudio.com/downloads/download-visual-studio-vs
   - SQL Server Express 2016 with Tools
     http://www.microsoft.com/en-us/download/details.aspx?id=42299
   - Web Browser (we prefer Google Chrome due to its extended development tools)
   - Git desktop client (we prefer either GitHub for Windows)
     https://github.com/

3) Personal requirements:
   - Professional e-mail account
     thorwisheshewasme@gmail.com,
     teddybearsrule@yahoo.com, and onehotnerd@hotmail.com
     are NOT professional email accounts
   - Professional resume (draft)
   - Github account https://github.com/
   - Microsoft online account https://www.microsoft.com/en-us/account
     (Use your professional e-mail mentioned above)

Course Description

This is a fast-paced, interactive learning experience that employs a learn-by-doing theory of education. Lectures are typically short in duration – 30 to 60 minutes – during which essential skills related to the next development project are taught. Lecture sessions are followed by coding sessions focused on solving real-world problems. Upon completion you will have completed in excess of 600 hours of coding during which you will have built multiple real- world projects, all of which will be made available for viewing by prospective employers on your personal Website.

You will be treated more as an "employee" of Coder Foundry than a student during your time in the course, with project specifications, weekly project deliverables, deadlines, and accountability interviews.

Mondays are reserved for accountability Interviews, during which you will report on the work you have completed during the previous week, your status on the current deliverable and project, and any pitfalls or difficulties you may have encountered. You will also demo your software in its current deliverable state and may be asked to discuss various aspects of the code you have written to that point. In addition, these interviews serve as an opportunity for us to coach you on valuable interviewing skills in an effort to better prepare you for the job interviews in which you will participate upon completion of the course.

Course Schedule

Week	Topics	Project(s)
1	Git Visual Studio Bootstrap JavaScript/jQuery Review Microsoft Azure	Bootstrap Exercises JavaScript Exercises Project: Personal Website
2	Relational Databases Entity Framework OOP with C# MVC - Code-First database development - Scaffolding - View models vs. data models - User management, role assignment - Authentication and authorization	Project: MVC #1
3	MVC - Querying databases with LINQ - Server-side paging	
4	MVC - Partial Views - Review and other topics More LINQ Project Development	Project: MVC #2
5	Project Development	
6	Project Development	
7	Project Development	
8	Relational Databases Review SQL Data modeling with SQL Server Management Studio Stored Procedures and advanced queries	Project: MVC #3 Database First Web API2 Mobile Cross-platform
9	Project Development	
10	Project Development	
11	Xamarin,Project Development More Stored Procedures Micro ORMs, Web API2	
12	Xamarin Project Development	

Note: We reserve the right to alter course schedules to best meet the needs of the class body, as determined by the instruction team.

Grading

We follow a grading pattern equivalent to that of an employer, as opposed to grading systems used in academic settings. All student projects will be assigned a Pass/Fail grade based upon strict adherence to project specifications, deliverables, and deadlines. Simply put, students who complete each project *on time and as required by the project specification* receive a passing grade for that project.

The course curriculum consists of six (6) total projects of varying degrees of difficulty.

1) JS/jQuery/HTML/CSS

2) Personal Web Site and Portfolio MVC - C# with HTML/CSS/JS/jQuery

3) MVC - C# with LINQ/Entity Framework/HTML/CSS/JS/jQuery

4) MVC - C# with LINQ/Entity Framework/HTML/CSS/JS/jQuery

5) Xamarin Forms with WebAPI2 & SQL

To receive a passing grade for the course, students must receive passing grades for all three (3) of the projects.

We actively work to help students *who receive passing grades* obtain employment by seeking out suitable job opportunities on their behalf, setting up job interviews, and providing interview coaching.

We cannot adequately market non-passing students to potential employers. Students must have a *suitable body of work* that we can show potential employers before we can schedule interviews. A suitable body of work is the result of a passing grade for the course.

Textbooks

We pride ourselves in our ability to sit at the leading edge of software development technologies, and do not make use of printed (i.e. quickly outdated) textbook materials. We do make use of a variety of online reference sources, which are updated from one course to the next as required.

Class Policies

All Coder Foundry students are expected to conduct themselves in a professional and respectful manner at all times. As a professional technology organization, we strongly adhere to the ACM/IEEE Code of Ethics and Professional Practice, and we expect the same conduct of our students. Students who knowingly and consistently act in a manner that violates or opposes this code of conduct will be dismissed from the course. In addition, the following policies are specific to Coder Foundry classroom operations.

Attendance

Coder Foundry is not designed to operate as an academic institution, but rather as a workplace simulation. Attendance is mandatory, just as it is on a job. The average company provides employees with two (2) weeks (ten (10) business days) of personal paid time off. We permit students four (4) total absences during the 12-week course. That's equivalent to 16 days of paid leave in the average job, far more than an employee would reasonably expect. Coder Foundry does not provide "make-up" opportunities for absences.

Students who accrue more than four (4) absences for any reason may be dropped from the course. Absences, regardless of reason, have no effect on project deadlines. A project that is incomplete or late receives a Did Not Pass grade.

Emergency situations such as serious illness or hospitalization, family deaths, and other extenuating circumstances are considered exceptions, and are dealt with on a case-by-case basis. Such occurrences require the presentation of documentation verifying the exceptional nature of the circumstance.

Tardiness

As on the job, tardiness is not acceptable. It disrupts the teacher as well as the other students and inhibits the learning environment. A student who is more than fifteen (15) minutes late for the day's session is considered tardy. Four (4) such occurrences will count as one (1) absence toward the student's allotted four (4) absences for the course. A student who arrives an hour or more late will be considered absent for the day.

Classroom Behavior

Disruptive behavior is unacceptable in any classroom and in any workplace. Students will not engage in non-class-related behavior, such as other employment activities, telephone conversations, video conferencing, online shopping, social networking, online videos or movies, video gaming or gambling, grooming, napping, or other behavior that is inappropriate to a work environment while in class. Students may be asked to leave for the day and considered absent should such behavior occur. Persistent behavior of this type will result in dismissal from the course. Dismissed students are not entitled to a refund of any portion of their tuition fees, as a dismissed student is not classified as a *withdrawn* student and is therefore not subject to the terms set forth for student refunds.

Respectful heckling of instructors is permitted in moderation.

Personal Integrity

Academic and professional integrity are of the utmost importance. Your work must be your own. While we encourage our students to assist and learn from one another, just as they would in a work environment, ultimately each student is responsible for his or her own work. Submitting a project that is someone else's work is absolutely unacceptable and will result in immediate dismissal from the course.

Students should understand that the coding profession is unlike that of a writer, in that it is common for coders to borrow from one another's solutions to specific coding problems. For example, if a student desires a solution for client-side pagination with AngularJS and is unsure how to derive such a solution independently, we encourage that student to research possible solutions. Doing so may lead the student to a coded solution on any one of a variety of resource websites. Borrowing others' coding solutions to specific problems is not plagiarism. Borrowing another person's project, or significant portion thereof, is. We encourage our students to use specific solutions that they discover through their own research, but also to endeavor to understand the solutions they choose to borrow. If a student cannot articulately explain his or her own code in a progress interview, that student will not pass the project.

Special Accommodations

If you have specific needs as the result of any disability, inform a member of the Coder Foundry staff before beginning the course so that necessary accommodations can be made.