Back-end Bootcamp **Course Overview**

Course Description

This project-based backend bootcamp provides hands-on experiences that introduces developers to the full web stack. Students will understand a day in the life of a backend developer while learning industry best practices. They will be introduced to programming concepts, databases, RESTful Web APIs and PHP development. This backend bootcamp guides and challenges new developers to become confident and proficient in debugging, refactoring, unit testing, design patterns, and security. Graduates will leave this bootcamp with tools, libraries, and resources that improve their skills and prepare them for the workforce as backend developers.

Target Audience:
Individuals with frontend web development experience interested in moving into a role as a backend developer. Individuals who possess basic knowledge of at least one programming language and who want to expand or modernize their skills.

Prerequisites:
WebLab Frontend Bootcamp, or equivalent experience; knowledge or experience with at least one programming language; introductory experience in software development; understands data types, loops functions; some experience in configuring any software

Competencies and Learning Objectives

Module 1: Intro and Overview

1. Explore the discipline of backend development
2. Describe the developer's mindset
3. Examine the landscape of backend development
   a. Version control
   b. Super editors and tools you use
   c. Command-line
4. Describe the day in the life of a backend developer
5. Develop a project scope
6. Utilize Readme Driven Development

Module 2: Programming Concepts and Debugging

1. Utilize primitive types: Boolean, integer, double, value types, strings, etc
2. Practice using language constructs
   a. Conditionals
   b. Loops
   c. Variables

d. Functions
3. Practice debugging
4. Detect common traps
5. Utilize best practices
6. Demonstrate how to step through the code
7. Utilize try-catch, exceptions

Module 3: Intro to Programming with PHP
1. Describe the difference between Server Side and Client Side Languages
2. Explore Methods of Execution
    a. Command Line
    b. Hypertext Processing
3. Explore PHP resources (php.net etc)
4. Explore basic PHP scripting
    a. Use PHP scripting basics
    b. Utilize PHP Syntax & Constructs
        i. Primitive data types
        ii. Constants and variables
        iii. Looping constructs
        iv. If/Else/switch/ternary
5. Explore more advanced PHP scripting
    a. Arrays, objects and resources
    b. Functions
        i. Arguments
        ii. Returning values
        iii. Recursion
        iv. Scope
    c. Practice using PHP built-in functions
6. Develop Forms (Get/Post)

Module 4: Intro to Databases/SQL
1. Examine basic database concepts
2. Explore database strengths, weaknesses and trade-offs
    a. Relational
    b. NoSQL, MongoDB, CouchDB, Redis
3. Practice using basic SQL syntax
4. Describe multi-tenancy
5. Examine when to use ORMs

Module 5: PHP Web Development (David)
1. Explore static vs dynamic content
2. Create an ORM with PHP
3. Explore sessions & authentication
4. Examine Cookies
5. Examine Object Oriented Programming
6. Discuss time/date operations in php
7. Read and write files
8. Build a dynamic database driven website

Module 6: Testing & Debugging
1. Utilize documentation driven development
2. Explore test driven development
3. Practice debugging
   a. Detect common traps
   b. Utilize best practices
   c. Demonstrate how to step through the code
   d. Utilize try-catch, exceptions
4. PHP error handling specifics

Module 7: Refactoring & Patterns
1. Practice Refactoring
   a. Why Refactor?
   b. What is Refactoring?
   c. Principles of Refactoring
   d. Code Smells to Refactor
   e. Refactoring Classes & Organization
   f. Refactoring Logic
2. Describe Types of Refactorings
   a. Moving features between objects
   b. Organizing data
   c. Simplifying conditional expressions
   d. Making method calls simpler
   e. Dealing with generalization
   f. Big refactorings
   g. Composing methods
3. Explore Principles of Refactoring to Patterns
   a. Describe a design pattern
   b. Explain why design patterns are used
   c. Describe the challenges that come with design patterns
   d. Demonstrate when to use patterns
4. Demonstrate Writing Loosely Coupled Code for Testing & Maintenance
   a. Describe best practices
   b. Demonstrate testing code dependent on databases and external services
   c. Leverage the separated interface pattern
   d. Practice TDD and Separated Interface

Module 8: Web APIs - Rest
1. Demonstrate use of PHP database interaction and send results to a client as json
2. Describe how JavaScript and Ajax make requests from client to server
3. Build your own API

Module 9: Security (in class discussion)
1. Authenticate the user to log in and out
2. Authorize the user to perform actions
3. Protect Against Vulnerabilities
    a. SQL Injection Attacks
    b. Cross-site Scripting Attacks

Module 10: Real World (All SMEs)
1. Examine how to make your life easier as a backend developer
    a. .NET Web API
    b. Ruby on Rails
    c. Ember.js
2. Compile useful libraries
3. Examine distributed computing and scaling
4. Examine MVC
5. Examine cloud computing

Integrated throughout: The Behaviors, Skills and Attitudes of a Successful Backend Developer
- ❏ Research skills
- ❏ Communication skills
- ❏ Care for code quality
- ❏ Teamwork and collaboration
- ❏ Problem solving and analytical skills
- ❏ Detail oriented
- ❏ Curious: self-learner, self-starter
- ❏ Creative thinking
- ❏ Taking initiative
- ❏   Flexibility

Updated 08/20/2015 at 12:39 PM