**Phần 4:**

**Events và Event Emitter trong NodeJS**

**Bài 1: Giới thiệu chung và mục tiêu của khóa học**

**Bài 2: Big Words và NodeJS**

**Bài 3: Khái niệm liên quan: Command Line Interface**

**Bài 4: Tài liệu tham khảo Command Line**

**Bài 5: Khái niệm Server và Client (Chủ - khách)**

**Bài 6: Phần nào trong Javascript cần để quản lý 1 Server?**

**Bài 7: Ngôn ngữ lập trình C++**

**Bài 8: Javascript**

**Bài 9: Tải source code mã nguồn chương trình cho khóa học**

**Bài 10: Cài đặt và chạy một số phần Javascript trong NodeJS**

**Bài 11: Khái niệm liên quan: Modules**

**Bài 12: Javascript: First-Class Functions và Function Expressions**

**Bài 13: Xây dựng một Modules**

**Bài 14: Javascript: Objects và Object Literals**

**Bài 15: Javascript: Prototypal Inheritance và Function Constructors**

**Bài 16: Javascript: By Reference và By Value**

**Bài 17: Javascript: Gọi Expressions Function (IIFEs)**

**Bài 18: Các Modules trong NodeJS thực sự hoạt động như thế nào? Sự khác nhau giữa "module.export" và "require"**

**Bài 19: Javascript: JSON**

**Bài 20: Tìm hiểu thêm về Require**

**Bài 21: Module Patterns**

**Bài 22: So sánh Exports với Modules**

**Bài 23: Yêu cầu Native (Core) Modules**

**Bài 24: Module và ES6**

**Bài 25: Web Server Checklist**

**Bài 26: Khái niệm Events**

**Bài 27: Javascript: Object Properties, First Class Function, và Arrays**

**Bài 28: Event Emitter - Phần 1**

**Bài 29: Event Emitter - Phần 2**

**Bài 30: Javascript: Object.create và Prototypes**

**Bài 31: Kế thừa từ Events Emitter - Phần 1**

**Bài 32: Javascript: Node, ES6, và Template Literals**

**Bài 33: Javascript: .call và .apply**

**Bài 34: Kế thừa từ Events Emitter - Phần 2**

**Bài 35: Javascript: ES6 Classes**

**Bài 36: Kế thừa từ Events Emitter - Phần 2**

**Bài 37: Javascript: Javascript là đồng bộ**

**Bài 38: Khái niệm Callbacks (Hàm hồi)**

**Bài 39: libuv, Events Loop và Non-Blocking Execution không đồng bộ**

**Bài 40: Khái niệm về Streams và Buffers**

**Bài 41: Dữ liệu nhị phân, Character Sets, và bảng mã.**

**Bài 42: Buffers (bộ đệm)**

**Bài 43: ES6 Typed Arrays**

**Bài 44: Callbacks trong Javascript**

**Bài 45: Files và fs**

**Bài 46: Streams**

**Bài 47: Khái niệm Pipes**

**Bài 48: Pipes**

**Bài 49: Web Server Checklist**

**Bài 50: Khái niệm TCP / IP**

**Bài 51: Khái niệm Addresses và Ports**

**Bài 52: Khái niệm HTTP**

**Bài 53: http\_parser**

**Bài 54: Sử dụng NodeJS để xây dựng 1 Web Server**

**Bài 55: Xuất ra files HTML và Templates**

**Bài 56: Streams và Performance**

**Bài 57: API và các thiết bị đầu cuối**

**Bài 58: Xuất ra JSON**

**Bài 59: Định tuyến**

**Bài 60: Web Server Checklist**

**Bài 61: Khái niệm Package và quản lý Package**

**Bài 62: Versioning Semantic (semver) là gì?**

**Bài 63: NPM và cách đăng ký NPM: Sử dụng mã của người khác**

**Bài 64: Init, nodemon, và package.json - Phần 1**

**Bài 65: NPM Global Installation**

**Bài 66: Init, nodemon, và package.json - Phần 2**

**Bài 67: Sử dụng mã của người khác**

**Bài 68: Cài đặt Express và xây dựng 1 Web Server hoàn chỉnh.**

**Bài 69: Routes**

**Bài 70: Static Files và Middleware**

**Bài 71: Templates và Template Engines**

**Bài 72: Querystring và Post Parameters**

**Bài 73: API RESTful và JSON**

**Bài 74: Cấu trúc 1 ứng dụng**

**Bài 75: Mối liên hệ giữa cơ sở dữ liệu và SQL**

**Bài 76: NodeJS và MySQL**

**Bài 77: NoSQL và Documents**

**Bài 78: MongoDB và Mongoose**

**Bài 79: Web Server Checklist**

**Bài 80: Yêu cầu phần mềm: node-todos**

**Bài 81: Khởi tạo ứng dụng**

**Bài 82: Cài đặt model dữ liệu với Mongodb và Mongoose**

**Bài 83: Khởi tạo dữ liệu**

**Bài 84: Tạo các API ứng dụng**

**Bài 85: AngularJS phía client: Mô hình Model-View-Star**

**Bài 86: Cài đặt AngularJS controller/app/view (p1)**

**Bài 87: Cài đặt AngularJS controller/app/view (p2)**

**Bài 88: Cài đặt AngularJS controller/app/view (p3)**

**Bài 89: Cài đặt AngularJS Services: Lấy dữ liệu từ phía Server**

**Bài 89 : Hoàn thiện và Deploy dự án lên Heroku**

**Bài 90 MongoDB, Express, AngularJS.**

**Bài 91 Tổng kết khóa học**