

## หลักสูตร MEAN Stack Developer

MEAN Stack คือ Framework ที่ใช้ในการจัดระบบเพื่อรองรับการพัฒนา Web Application ในรูปแบบทั้ง Front-End และ Back-End ประกอบด้วย 4 ส่วนหลักคือ MongoDB Express Angular และ Node.js ซึ่งการใช้งานดังกล่าวจะเน้นไปที่ตัวภาษา JavaScript ซึ่งผู้ใช้จะต้องมีความรู้พื้นฐานในเรื่องนี้ เพราะ JavaScript เป็นภาษาที่ถูกใช้มากที่สุดในโลกภาษาหนึ่ง และสำหรับ NodeJS นั้นก็เป็นภาษาที่ง่ายต่อการเรียนรู้ สะดวกในการต่อยอด อีกทั้งยังมี community ขนาดใหญ่รองรับในการใช้งาน จึงเป็นภาษาที่เหมาะสมในการเรียนรู้สำหรับเพิ่มศักยภาพของกลุ่มคนอายุ 35 ปีขึ้นไป (Mid-Career) และในปัจจุบันมีความสามารถในการใช้งานได้หลากหลายรูปแบบ

### เนื้อหาการอบรม

วันที่	เนื้อหาการอบรม
วันที่ 1	<p><b>Basic Node.js</b></p> <p>พื้นฐานการพัฒนาโปรแกรมด้วย JavaScript</p> <ul style="list-style-type: none"> <li>• Code structure</li> <li>• Variables</li> <li>• Data types</li> <li>• Basic operators, math</li> <li>• Comparisons</li> <li>• Conditional branching</li> <li>• Logical operators</li> <li>• Loops: while and for</li> <li>• The switch statements</li> <li>• Functions</li> <li>• Arrow functions</li> </ul> <p>ความรู้เบื้องต้นเกี่ยวกับ Node.js</p> <ul style="list-style-type: none"> <li>• ประวัติ ความเป็นมา สถานการณ์ที่เหมาะสมแก่การใช้งาน</li> <li>• การติดตั้ง (Installation)</li> </ul> <p>Node.js การใช้งาน Core APIs</p> <ul style="list-style-type: none"> <li>• การติดตั้งและเรียกใช้ Modules</li> <li>• การประมวลผลข้อมูลแบบชนิดต่าง ๆ (Data Manipulation) เช่น ข้อมูลแบบ Text, Binary, และ JSON เป็นต้น</li> </ul> <p>TypeScript เบื้องต้น</p> <ul style="list-style-type: none"> <li>• การพัฒนาโปรแกรมด้วย TypeScript</li> <li>• การ compile TypeScript ให้เป็น JavaScript</li> </ul>

วันที่	เนื้อหาการอบรม
	<p>การบริหารการประมวลผลแบบ Asynchronous (Asynchronous Control Flow)</p> <ul style="list-style-type: none"> <li>• แบบแผนการทำงาน (Asynchronous Pattern)</li> <li>• การใช้งาน Asynchronous Control Flow Module</li> </ul>
วันที่ 2	<p>การสร้าง web application ด้วย NestJS Framework NestJS framework</p> <ul style="list-style-type: none"> <li>• การติดตั้ง และการเตรียมการ</li> <li>• การใช้งาน Web Template</li> <li>• การใช้งาน URL Routing</li> <li>• การใช้งาน Simple Route Middleware</li> <li>• การใช้งาน JWT ร่วมกับ Nest.js</li> </ul> <p>การติดตั้ง และการเตรียมการ</p> <ul style="list-style-type: none"> <li>• การเขียน node ติดต่อ MySQL ด้วย TypeORM</li> </ul> <p>Automate Unit Test</p> <ul style="list-style-type: none"> <li>• การติดตั้ง jest framework</li> <li>• การเขียน unit test ด้วย jest</li> </ul>
วันที่ 3	<p><b>พัฒนา REST API ด้วย NodeJS</b></p> <ul style="list-style-type: none"> <li>• วิธีการพัฒนา REST API ด้วย Node.js ร่วมกัน Framework NestJS</li> <li>• วิธีการเชื่อมต่อฐานข้อมูล My SQL ด้วย NodeJS</li> <li>• วิธีการพัฒนา CRUD, Pagination, Filter ด้วย NodeJS</li> <li>• มาทำความรู้จักกับ CORS และแนวทางการแก้ปัญหาใน NodeJS</li> <li>• ทำ Workshop พัฒนา REST API จัดการข้อมูล</li> </ul>
วันที่ 4	<p><b>Angular Framework</b></p> <p><b>Module 1: Architecture Overview</b></p> <ul style="list-style-type: none"> <li>• Basics of TypeScript</li> <li>• Components, Bootstrap, and the DOM</li> <li>• Directives and pipes</li> <li>• Data binding</li> </ul> <p><b>Module 2: Components</b></p> <ul style="list-style-type: none"> <li>• Component metadata</li> <li>• The component selector</li> <li>• The component templates</li> <li>• Property binding</li> </ul>

วันที่	เนื้อหาการอบรม
	<ul style="list-style-type: none"> <li>● Getting data to the component with input</li> <li>● Subscribing to component events with output</li> </ul> <p><b>Module 3: Directives and Pipes</b></p> <ul style="list-style-type: none"> <li>● Structural directives – ngIf</li> <li>● Structural directives – ngFor</li> </ul> <p><b>Module 4: Directives and Pipes</b></p> <ul style="list-style-type: none"> <li>● Using directive values</li> <li>● Working with events in directives</li> <li>● Angular pipes – built in</li> <li>● Angular pipes – custom</li> </ul> <p><b>Module 5: Form</b></p> <ul style="list-style-type: none"> <li>● Angular Form</li> <li>● Template – driven forms</li> <li>● Model – driven forms</li> <li>● Validation – built in</li> <li>● Error handling</li> </ul>
วันที่ 5	<p><b>Angular Framework #2</b></p> <p><b>Module 6: Dependency Injection and Services</b></p> <ul style="list-style-type: none"> <li>● How Angular does dependency injection</li> <li>● Services in Angular</li> <li>● Building Class constructor injection</li> <li>● a service</li> </ul> <p><b>Module 7: HTTP</b></p> <ul style="list-style-type: none"> <li>● The Angular 6 HTTP bundle</li> <li>● Using HTTP for GET calls</li> <li>● Using HTTP for POST, PUT, and DELETE calls</li> </ul> <p><b>Module 8: Routing</b></p> <ul style="list-style-type: none"> <li>● The Angular 6 routing bundle</li> <li>● Route configuration</li> <li>● Router outlets</li> <li>● Router links</li> </ul>
วันที่ 6	พัฒนา Web Application ด้วย Angular

วันที่	เนื้อหาการอบรม
	<ul style="list-style-type: none"> <li>● เรียนรู้วิธีการดึงข้อมูลจาก NodeJS ผ่าน HTTP Client</li> <li>● เรียนรู้วิธีการนำข้อมูลที่ได้จาก API มาใช้งาน</li> <li>● เรียนรู้วิธีการส่งข้อมูลจาก Angular ไปยัง NodeJS</li> <li>● ทำ Workshop พัฒนาโปรแกรมเพื่อเชื่อมโยงข้อมูลกับ NodeJS ผ่าน API</li> <li>● เรียนรู้วิธีการ Deploy โปรแกรมที่พัฒนาเสร็จแล้วด้วยวิธีการต่าง ๆ อันได้แก่             <ul style="list-style-type: none"> <li>○ Deploy ด้วย Docker</li> <li>○ Deploy ด้วย pm2</li> </ul> </li> </ul>