**NodeJs Micro Service Curriculum**

1. **Introduction to Node.js**

Learning Objective: In this module, you will learn what is Node.js and what makes Node.js so

popular. You will also learn how to use Node Package Manager (NPM) and Nodemon.

**Topics:**

What is Node.js?

Why Node.js?

Installing NodeJS

Node in-built packages (buffer, fs, http, os, path, util, url)

Node Modules

Import your own Package

Node Package Manager (NPM)

Local and Global Packages

Push code to GitHub

1. **File System Module and Express.js**

Learning Objective: In this module, you will learn how to get user inputs via Command Line

Arguments and store data using the File system. You will also learn how to create the applications

using Express Framework, whereas manage and deploy them using PM2 and Nginx.

Topics:

Get Input from Users

Pass Multiple Arguments with Yargs

File System Module

Operations associated with File System Module

JSON Data

Http Server and Client

Express Framework

Run a Web Server using Express Framework

Routes

Deploy application using PM2 and Nginx

1. **Asynchronous Programming**

Learning Objective: In this module, you will learn how to develop asynchronous Node.js application

using the Call stack, Callback queue, and Event Loop mechanism. You will also gain knowledge on how

to work with HTTP requests, Promises, and EJS templates.

Topics:

Call Stack

Callbacks, Callback Queue and Event Loop

Callback Abstraction

Callback Chaining

Promises

Promise Chaining

Request Package

Customizing HTTP Requests

Error handing with appropriate HTTP codes

Introduction to template engine (EJS)

Integration with MongoDB and Email Servers

Learning Objective: In this module, you will learn how to work with NoSQL Database – MongoDB.

Also, you will learn how to send emails via Node.js application.

**Topics:**

Introduction to NoSQL Databases and MongoDB

Installation of MongoDB on Windows

Installation of Database GUI Viewer

Inserting Documents

Querying, Updating and Deleting Documents

Connect MongoDB and Node.js Application

Exploring SendGrid

Sending emails through Node.js application using SendGrid

1. **REST APIs and GraphQL**

Learning Objective: In this module, you will learn how to run queries via Node.js application along

with some important aspects of Node.js like REST APIs, setting up Mongoose, Postman Installation and

GraphQL.

Topics:

Structuring a REST API

Separate Route Files

Resource Creation Endpoints

Resource Reading Endpoints

Resource Updating Endpoints

Resource Deleting Endpoints

Postman Installation

Setting up Mongoose

Creating a Mongoose Model

Creation of basic query

Reading data from the array

Calling API

Query building using API Mutation

GraphQL Playground

1. **Building Node.js Applications using ES6**

Learning Objective: In this module, you will learn how to write your application using ES6. Also, you

will learn how to Design Dashboard and perform CRUD operations.

Topics:

History of ES6

Difference between ES5 and ES6

ES6 Functions

Object Manipulation with ES6

Import and Export with ES6

Template Literal in ES6

Async/Await

Transpilation

Introduction to Babel

Creating an API using ES6

Data Validation and Sanitization

Creating Dashboard Application with ES6 syntax and EJS Template

1. **Testing Node.js Applications**

Learning Objective: In this module, you will learn how to test your Node.js applications using Mocha

and Chai, along with some important testing concepts related to Node.js Application.

Topics:

Writing Tests and Assertions

Testing Asynchronous Code

Testing an Express Application

Setup and Teardown

Testing with Authentication

Advanced Assertions

Mocking Libraries

Wrapping up User Tests

Setup Task Test Suite

Testing with Task Data

1. **Microservices Application**

Learning Objective: In this module, you will learn how to build a Microservices based Web

application with Node.js and Deploy it to Docker Container.

Topics:

Microservices

Child Process

Docker

Introduction to Docker Hub

Advantages of Microservices with Node.js

Create a microservices application using Node.js and deploy it to docker container

1. **Project**

How will I execute the practicals?

To execute the practical’s, you will require an environment for Node.js and MongoDB. All Node.js

applications will be built and debugged using IDE Visual Studio Code. The detailed step by step

installation guides is available on LMS. In case you come across any doubt, the 24\*7 support team will

promptly assist you.