Marwadi University

BACKEND DEVELOPMENT

Node JS
Express JS
MongoDB, Mongoose

Instructor 1   : **Harsh Siddhapura** *(SDE)*

Instructor 2   : **Harshal Faldu** *(SDE)*

Head of Dept.   : **Prof. Chandrasinh Parmar**

Department   : **Information & Communication Technology**

University   : **Marwadi University, Rajkot**

Table of Content

1. Course Plan: 1 Month

❖ **Assignments/Project : 20 Hours**
  ➢ Monday   : 1 Hour * 4 Weeks   = 4 hours
  ➢ Tuesday   : 1 Hour * 4 Weeks   = 4 hours
  ➢ Wednesday : 1 Hour * 4 Weeks   = 4 hours
  ➢ Thursday   : 1 Hour * 4 Weeks   = 4 hours
  ➢ Friday   : 1 Hour * 4 Weeks   = 4 hours

❖ **Doubt Solving Sessions : 8 Hours (Evening)**
  ➢ Friday : 2 Hours * 4 Weeks = 8 Hours

❖ **Lectures : 24 Hours (Morning)**
  ➢ Saturday : 3 Hours * 4 Weeks   = 12 Hours
  ➢ Sunday   : 3 Hours * 4 Weeks   = 12 Hours

● **Total : 52 Hours**

● Week 1 : Git + Javascript + Assignments
● Week 2 : Node JS + Third Party APIs + Assignments
● Week 3 : Express JS + MVC + Group Project
● Week 4 : MongoDB + Mongoose + Group Project

2. Setup & Basics

❖ Setup
  ➢ Visual Studio Code : https://code.visualstudio.com/download
  ➢ Node JS : https://nodejs.org/en/download/
  ➢ Postman : https://www.postman.com/downloads/
  ➢ Git : https://git-scm.com/downloads

3. Git Basics

❖ Git Basics: https://education.github.com/git-cheat-sheet-education.pdf

❖ Session Notes:
  ➢ git config --global user.name "[firstname lastname]"
  ➢ git config --global user.email "[valid-email]"
  ➢ git clone [url] .
  ➢ git status
  ➢ git add [file]
  ➢ git commit -m "[descriptive message]"

4. Javascript Basics

❖ Javascript Basics :
  ➢ Learning: https://www.w3schools.com/js/
  ➢ Practice: https://the-winter.github.io/codingjs/

Topics:

❖ JS Tutorial

5. Introduction to Node.JS

❖ **What is npm?**

npm is two things:
- first and foremost, it is an online repository for the publishing of open-source Node.js projects
- second, it is a command-line utility for interacting with a said repository that aids in package installation, version management, and dependency management.

Many Node.js libraries and applications are published on npm, and many more are added daily. These applications can be searched for on https://www.npmjs.com/. Once you have a package you want to install, it can be installed with a single command-line command.

❖ **What is Node.js?**
  - ➢ Node.js is an open-source server environment
  - ➢ Node.js is free
  - ➢ Node.js runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
  - ➢ Node.js uses JavaScript on the server

❖ **Why Node.js?**

A common task for a web server can be to open a file on the server and return the content to the client.

- Here is how PHP or ASP handles a file request:

  - ➢ Sends the task to the computer's file system.
  - ➢ Waits while the file system opens and reads the file.
  - ➢ Returns the content to the client.
  - ➢ Ready to handle the subsequent request.

- Here is how Node.js handles a file request:

  - ➢ Sends the task to the computer's file system.
  - ➢ Ready to take the next request.
  - ➢ When the file system has opened and read the file, the server returns the content to the client.
  - ➢ Node.js eliminates the waiting and simply continues with the next request.

Node.js runs single-threaded, non-blocking, asynchronous programming, which is memory efficient.

❖ **What Can Node.js Do?**
  ➢ Node.js can generate the dynamic page content
  ➢ Node.js can create, open, read, write, delete, and close files on the server
  ➢ Node.js can collect form data
  ➢ Node.js can add, delete, and modify data in your database

❖ Core module in Node js
  ➢ Global Modules
    ■ console
  ➢ Non-Global Modules
    ■ fs, Buffer

❖ Make basic server in node js
❖ What are the package.json file and use in node js
❖ What is Nodemon in node js
❖ Make Simple API in node js
❖ Input from the command line
❖ Show file List with file system
❖ CRUD with file system
❖ Asynchronous Node js
❖ Handle Asynchronous Data in Node js
❖ How Node js Works

6. Introduction to Express JS

❖ Introduction of Express js
❖ Introduction to MVC structure
❖ Express js folder structure and its explanations
❖ Routing, Rendering, Middlewares, Templating etc
❖ EJS file format
❖ Cookies & Sessions
❖ Debugging
❖ Error Handling
❖ Request type and its usage
❖ Project allocation and structure setup

7. MongoDB and Mongoose

❖ Introduction and basis of MongoDB
❖ MongoDB compass setup and guide
❖ CRUD operations
❖ Introduction of Mongoose package
❖ Connecting with Database
❖ Database operations (find, findOne, aggregate etc)
❖ Indexing in database
❖ Hands-on practice
❖ Other database concepts (If possible and if required)

- Thank You...

References

[1] https://www.w3schools.com/nodejs/default.asp

[2] https://www.tutorialspoint.com/expressjs/index.htm