Timing: *5:30 PM – 3:30 AM*

Backend Basics & Setting Up Your Development Environment

5:30 PM – 6:00 PM | Introduction: What is Backend Development?

- **What is Backend?** The “engine room” of your website that stores data, processes requests, and powers your app behind the scenes.
- **Why it matters:** Your frontend collects emails, but backend stores them, verifies emails, sends campaigns, etc.
- **Analogy:** Frontend = “dashboard and buttons” users see, backend = “workshop” where data is made ready.
- **Visual aid:** Picture a restaurant: the frontend is the waiter; backend is the kitchen.

6:00 PM – 7:00 PM | Installing Node.js on Windows

- **What is Node.js?** Node.js lets you run JavaScript on your computer to build backend services.
- **Why Node.js?** Frontend developers already know JS, so this is the easiest way to get the backend running.

Steps:

- Go to https://nodejs.org/en/download
- Click Windows Installer (.msi) for the LTS version
- Run the installer and follow prompts (Accept license, Next, install)
- After install, open **Command Prompt**:
    - Press Win + R, type cmd, hit Enter
    - Type

- You should see the installed versions printed

7:00 PM – 7:30 PM | Installing Visual Studio Code (VS Code) on Windows

- **Download VS Code installer from https://code.visualstudio.com/**
- **Run the installer, check options like "Add to PATH" and "Open with Code" context menu for convenience**
- **Open VS Code**
- **Open an integrated terminal in VS Code by pressing: Ctrl + \ (or View > Terminal)**
- **This terminal uses PowerShell or Command Prompt**

7:30 PM – 8:30 PM | Installing Postman (API Testing Tool)

- **What is Postman?** A tool to test backend APIs without writing frontend code.
- **Why Postman?** To check if your backend works correctly by sending requests manually.

Steps:

- Download Postman Windows installer from https://www.postman.com/downloads/
- Run installer and open Postman
- Explore UI, learn how to create a new request (this is for testing backend APIs)

8:30 PM – 9:00 PM | Break

9:00 PM – 10:30 PM | Installing Git & Introduction to Version Control

- **What is Git?** A tool to save your code and track changes — so you don't lose work.
- **Why Git?** Important for teamwork, and good development practice.

Steps:

- Download Git for Windows installer from https://git-scm.com/download/win
- Run the installer, select options (use default options unless you want to customize; important: select "Git from the command line and also from 3rd-party software")
- After install, open **Git Bash** (installed with Git) from Start Menu or right-click inside a folder > "Git Bash Here"
- Check version:

```bash
git --version
```

- Set your user name and email:

```bash
git config --global user.name "Your Name"
git config --global user.email "your.email@example.com"
```

10:30 PM – 12:00 AM | Create Your First Backend Project with Express

Steps:

- **Open File Explorer, create a folder called my-backend-project in your Documents or Desktop**
- **Open VS Code**
- **Open the folder my-backend-project in VS Code (File > Open Folder)**
- **Open integrated terminal (`Ctrl + ``) inside VS Code**
- **Initialize npm:**

```bash
npm init -y
```

- **Install Express:**

```bash
npm install express
```

- **Create `index.js` file and paste this code:**

```js
const express = require('express');
const app = express();

app.get('/', (req, res) => {
  res.send('Hello from your first backend server on Windows!');
});

app.listen(3000, () => {
  console.log('Server running at http://localhost:3000');
});
```

- **Run the server:**

```bash
node index.js
```

- **Open browser, visit http://localhost:3000**

12:00 AM – 12:30 AM | Dinner Break

12:30 AM – 2:00 AM | Workshop: Add More Routes and Test Them

- Add new route in `index.js`:

```js
app.get('/greet', (req, res) => {
  res.json({ message: 'Welcome to backend training on Windows!' });
});
```

- Restart the server (Ctrl + C to stop, then node index.js again)
- Test with browser and Postman

2:00 AM – 3:00 AM | Practice Task: Build Contact Info Routes

- Add two routes:

```js
app.get('/contact', (req, res) => {
  res.send('Contact page - Windows test');
});

app.get('/about', (req, res) => {
  res.json({ project: 'Email Marketing Backend', platform: 'Windows' });
});
```

- Test these routes in browser and Postman
- Modify responses to try different messages
- Restart server after changes

3:00 AM – 3:30 AM | Wrap-Up & Q/A

- Recap today's learning
- How to start server next time

- Common issues and how to fix
- Preview of tomorrow: REST API basics & working with JSON

Useful Links & Videos for Day 1

- Node.js Installation Guide (Video)
- Express.js Hello World Tutorial (https://expressjs.com/en/starter/hello-world.html)
- Postman Beginner Tutorial
- Git for Beginners
- VS Code Quickstart (https://code.visualstudio.com/docs/getstarted/introvideos)