Introduction

Web development doesn't stop at the front end. To create fully functional applications, you need to understand how servers, databases, and APIs come together. Node.js has revolutionized backend development by allowing developers to use JavaScript on both the client and server. Express.js builds on this power, providing a fast, minimal, and flexible framework to create scalable server-side applications.

This ebook is your step-by-step guide to mastering backend development with Node.js and Express. You'll not only learn the fundamentals but also apply them by building real-world projects that mirror today's industry standards.

What You'll Learn

- **Node.js Fundamentals** – Understand asynchronous JavaScript, the event loop, and how Node.js handles requests.

- **Express.js Basics** – Build your first server, define routes, and manage middleware.

- **Working with Databases** – Learn how to connect Node.js to MongoDB and PostgreSQL.

- **Authentication & Security** – Implement JWT-based authentication and secure your APIs.

- **RESTful APIs** – Design and deploy robust APIs for modern applications.

- **Deployment** – Take your project live on platforms like Vercel, Render, or AWS.

Who This Book Is For

- Frontend developers who want to become full-stack engineers.

- Beginners looking to break into backend development.

- Anyone curious about how real-world applications handle data, authentication, and scalability.

Sneak Peek: Your First Express App

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

app.get('/', (req, res) => {
  res.send('Hello, DigiPort Developer!');
});

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

With just a few lines of code, you've created your first web server! By the end of this book, you'll be building full-fledged applications with authentication, databases, and APIs—all ready for deployment.

✅ **By the time you finish this ebook, you'll have the skills and confidence to build and deploy full-stack applications with Node.js and Express.**