Backend Development Internship Weekly Report

This report summarizes the weekly progress and topics covered during the backend development internship. The focus throughout the internship was on strengthening backend development fundamentals using JavaScript, Node.js, Express, databases, real-time systems, logging, and microservice concepts.

WEEK 1: JavaScript Fundamentals & Advanced Concepts

- Variables and Datatypes
  - Learned primitive types (string, number, boolean, null, undefined).
  - Explored non-primitive types (objects, arrays).
- Arrays
  - Practiced array methods (map, filter, reduce).
  - Sorting, searching, iteration techniques.
- Operators
  - Basic, comparison, logical, and bitwise operators.
- Functions
  - Regular functions, arrow functions, shorthand syntax.
- Loops
  - For loop, while loop, loop control.
- Conditions
  - If-else, switch-case.
- Asynchronous JavaScript
  - Callbacks, Promises, Async/Await.
- File System
  - Performed read/write operations using fs module.

WEEK 2: Node.js Backend Development

• Environment Setup

- Used VS Code as IDE.

• Understanding Node.js

- Internal working (event loop, single-threaded architecture).
- Strict mode in Node.js.
- Concept of modules.

• Package Management

- Learned npm, package.json, script management.

• Modules & File Structure

- ES Modules vs CommonJS.
- Created and imported modules.
- Organized backend projects in folders.

• Version Control

- Used Git for source control.
- Managed repositories on GitHub.

WEEK 3-4: REST API Development (Express.js)

• HTTP methods and routing.

• Middleware concepts.

• View engine basics.

• Passport.js research for authentication.

• Scheduler / CRON jobs

- Cron patterns and use cases.

• Databases

- SQL & NoSQL differences.

- Sequelize ORM (models, migrations, CRUD).

- Mongoose basics.

WEEK 5: WebSockets & Developer Utilities

• Web Sockets

- How sockets work.

- Socket client vs server.

- middleware.

• API Documentation

- Markdown documentation.

- Swagger UI for REST API documentation.

WEEK 3-6: TASK IMPLEMENTATION

Task 1: Task Reminder System

• Developed using Node.js + JavaScript.

• Standard folder structure followed.

• Sequelize migrations used for database setup.

• CRUD APIs documented using Swagger.

• GitHub repository maintained.

Task 2: Simple Chat Application

• Real-time chat using Socket.IO.

• Implemented rooms and middleware.

• Integrated UI using a view engine or React.

THIS WEEK : Microservices & gRPC

• Microservice architecture

- Compared monolithic vs microservice.

- Understood benefits and trade-offs.

• Protocol Buffers

- .proto files, Protobuf vs JSON.

• gRPC

- Built sample gRPC client and server.

CONCLUSION

Throughout the internship, I consistently learned and implemented backend development concepts. Although the roadmap included TypeScript, I completed the entire backend workflow specifically using JavaScript. The internship enhanced my backend development skills, improved my understanding of real-world backend systems, and strengthened my development workflow using Git, database migrations, sockets, and microservices.