Full Stack Development with MERN
API Development and Integration Report

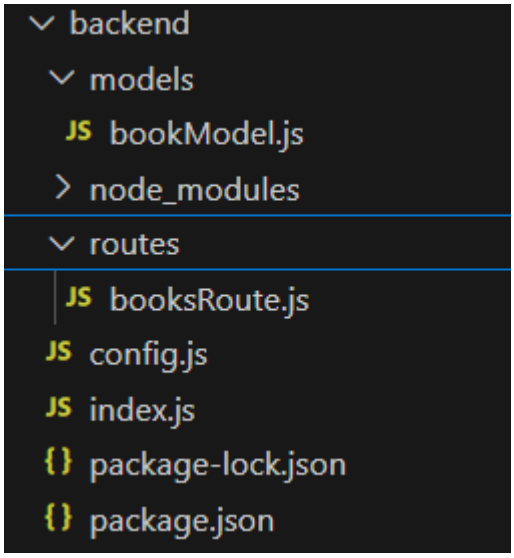
Date	22 June 2024
Team ID	SWTID1720433526
Project Name	Project - Book-Store
Maximum Marks	12 Marks

Project Title: Book-Store
Date: 19-07-2024
Prepared by: Matthew Jeannot Aranjani

Objective
The objective of this report is to document the API development progress and key aspects of the backend services implementation for the Book-Store project.

Technologies Used
Backend Framework: Node.js with Express.js
Database: MongoDB

Project Structure:



models - It is the folder containing the models.
bookModel.js - Defines the schema and model for the model for the books in the database.
routes - It is the folder containing the routes.
booksRoute.js - Define the routes for handling book-related API requests.
config.js - Contains the configuration settings to connect to MongoDB database.
Index.js - The main entry point of the application that sets up the server and connects all the components.

Key Directories and Files

1. **/models** - Includes Mongoose schemas and models for MongoDB collections.
2. /**routes** - Defines the API endpoints and links them to controller functions.
3. **/config** - Configuration files for database connections, environment variables, etc.

API Endpoints
A summary of the main API endpoints and thier purposes:

post / - Saving a new book to database
get / - Get all books from database
get/:id - Get one book from database
put/:id - Update a book
delete/:id - Delete a book

Integration with Frontend
The backend communicates with the frontend via RESTful APIs. Key points of integration include:
Data Fetching: Frontend components make API calls to fetch necessary data for display and interaction.

Error Handling and Validation
Describe the error handling strategy and validation mechanisms:
Error Handling: Centralized error handlign using middleware.