DevifyX Node.js Assignment

Assignment Title: User Activity Logger

Assignment Description: Store actions like login & views

Assignment Deadline: 7 Days

Objective

Develop a robust backend-only Node.js application to log and manage user activities (such as login and page views) via a RESTful API. No frontend implementation is required.

Core Features

1. **User Authentication:** Implement secure user registration and login using JWT-based authentication.
2. **Activity Logging:** Log user actions such as login, logout, and page views with timestamps.
3. **API Endpoints:** Provide RESTful API endpoints for logging and retrieving user activities.
4. **Activity Filtering:** Allow filtering of activities by user, action type, and date range.
5. **Pagination:** Support pagination for activity retrieval endpoints.
6. **Error Handling:** Implement comprehensive error handling for all API endpoints.
7. **Input Validation:** Validate all incoming request data for security and integrity.
8. **Database Integration:** Use a database (MongoDB or PostgreSQL) to persist user and activity data.

Bonus Features

- Real-time activity updates using WebSockets.
- Support for additional activity types (e.g., password changes, profile updates).
- Role-based access control for activity retrieval.
- Integration with external logging/monitoring services.

Technical Requirements

- Node.js (v14 or above)
- Express.js framework
- MongoDB or PostgreSQL as the database
- JWT for authentication
- Proper use of environment variables
- Well-structured project architecture (MVC or similar)
- Use of Git for version control
- Clear and concise API documentation (Swagger or similar)

Deliverables

- Source code repository (GitHub or similar)
- README file with setup instructions
- API documentation (Swagger/OpenAPI or Markdown)
- Sample environment file (`.env.example`)
- Database schema or migration scripts
- Postman collection (optional)

Use of AI Tools

> You are **permitted and encouraged** to use AI-based coding tools such as **GitHub Copilot**, **ChatGPT**, or similar platforms to assist with code generation, debugging, and documentation. However, the final submission should reflect your own understanding and structure.

Submission

Submit your assignment using the following form:
`https://forms.gle/LAvLWFmHRLXswwsx5`

Evaluation Criteria

- **Correctness:** Implementation meets the requirements and handles edge cases.
- **Code Quality:** Clean, readable, and maintainable code with proper structure.
- **Security:** Proper authentication, validation, and error handling.
- **Documentation:** Clear setup instructions and comprehensive API documentation.
- **Bonus Features:** Implementation of any additional features.
- **Timely Submission:** Assignment submitted within the deadline.

Click here to read our Terms and Conditions