Backend Developer Technical Challenge

School Management System API

Dear Candidate,

As part of our evaluation process, we invite you to participate in a technical challenge designed to assess your backend development capabilities. This challenge will help us better understand your approach to system design, code organization, and problem-solving skills.

Challenge Overview

You are tasked with developing a School Management System API using our standard project template available at: https://github.com/qantra-io/axion. Please maintain the existing architectural patterns and structure of the template.

Core Requirements

System Architecture

- Develop a RESTful API service using JavaScript
- Implement role-based access control (RBAC)
- Utilize either MongoDB or Redis for data persistence
- Follow the existing project template structure

Key Entities

1. **Schools**
    - Managed by superadmins
    - Complete CRUD operations
    - School profile management

2. **Classrooms**
    - Managed by school administrators
    - Associated with specific schools
    - Capacity and resource management
3. **Students**
    - Managed by school administrators
    - Enrollment and transfer capabilities
    - Student profile management

Technical Requirements

- Implement comprehensive input validation
- Include error handling and appropriate HTTP status codes
- Design and implement proper database schema
- Create authentication and authorization middleware
- Follow RESTful API best practices
- Include API rate limiting and security measures

Authentication & Authorization

- Implement JWT-based authentication
- Role-based permissions:
    - Superadmin: Full system access
    - School Administrator: School-specific access
    - Limited to their assigned school's resources

Deliverables

Required Components

1. Fully functional API implementation
2. Comprehensive API documentation
3. Database schema design

4. Test cases and results
5. Deployment instructions

Documentation

- API endpoint specifications
- Request/response formats
- Authentication flow
- Error codes and handling
- Database schema diagram

Evaluation Criteria

- Code quality and organization
- Security implementation
- API design and documentation
- Database design
- Error handling
- Testing coverage
- Performance considerations

Submission Guidelines

1. Host the application on a public hosting service
2. Upload code to a public repository
3. Include setup instructions in README.md
4. Complete the submission form with the following:
    - Repository URL
    - Deployed application URL
    - Any additional notes or considerations

Timeline

Please submit your completed challenge within 3 calendar days of receiving this email. The deadline for submission is [Date + 3 days] at 11:59 PM (Local Time).

Notes

- Feel free to make architectural decisions while maintaining the template structure
- Include any assumptions made during development in your documentation
- Consider scalability and maintainability in your implementation

We look forward to reviewing your submission. If you have any questions or need clarification, please don't hesitate to reach out.

Best regards,
Technical Recruitment Team