Technical Report: Class Booking API

1. Project Overview

Project Title

Class Booking API - Complete Backend System for Fitness Class Management

Project Description

A robust NestJS backend API designed for fitness centers and gyms to manage class bookings, user registrations, and credit-based payments with comprehensive business rules validation.

Core Objectives

- ✅ User authentication and authorization
- ✅ Class management with scheduling
- ✅ Credit-based booking system
- ✅ Business rules enforcement
- ✅ Real-time monitoring and metrics
- ✅ Comprehensive API documentation

2. Technical Architecture

Technology Stack

Component	*Technology*	*Purpose*
Framework	NestJS	Scalable backend architecture
Database	MongoDB + Mongoose	Flexible data storage
Authentication	JWT + Passport	Secure user management
Validation	class-validator	Request validation
Documentation	Swagger/OpenAPI	API documentation
Testing	Jest	Unit and integration tests

3. Key Features Implemented

🔐 Authentication System

- JWT-based registration and login
- Password hashing with bcrypt
- Protected routes with guards
- Token expiration handling

👥 User Management

- User registration and profile management
- Credit system for payments
- Role-based access control
- Secure password handling

Class Management

- Complete CRUD operations
- Instructor scheduling with overlap prevention
- Participant tracking
- Capacity management

Booking System

- Credit-based booking with validation
- Overlapping class prevention
- Cancellation with refund policies
- Real-time availability checking

Monitoring & Health

- Real-time health checks
- Performance metrics with Prometheus
- Database connection monitoring
- Memory and uptime tracking

Database Relationships

- **One-to-Many**: User → Bookings
- **One-to-Many**: Class → Bookings
- **Many-to-Many**: Class ↔ Users (through participants)

5. Business Logic & Validation

Booking Rules

1. **Credit Validation**: User must have sufficient credits
2. **Capacity Check**: Class must have available spots
3. **Time Conflict Prevention**: No overlapping bookings
4. **Duplicate Prevention**: Cannot book same class twice

Cancellation Rules

1. **Time Limit**: Cancellation allowed up to 2 hours before class
2. **Credit Refund**: Full refund for timely cancellations
3. **No Refund**: Within 2 hours of class start

Class Management Rules

1. **Instructor Availability**: No overlapping classes for same instructor
2. **Participant Safety**: Cannot delete classes with active bookings
3. **Capacity Management**: Automatic participant counting

7. Security Implementation

Authentication & Authorization

- JWT token-based authentication
- Password hashing with salt rounds
- Route protection with guards
- Role-based access control

Data Validation

- Input validation with class-validator
- MongoDB injection prevention
- XSS protection
- CORS configuration

Security Headers

- Helmet.js for security headers
- Rate limiting implementation
- SQL injection prevention
- Data sanitization

9. Conclusion

Project Success Metrics

- ✅ **Functional Requirements**: All core features implemented
- ✅ **Code Quality**: Comprehensive testing and validation
- ✅ **Performance**: Optimized database queries and API responses
- ✅ **Security**: Robust authentication and data protection
- ✅ **Documentation**: Complete API documentation

Technical Achievement

The Class Booking API successfully demonstrates modern backend development practices with NestJS, featuring a scalable architecture, comprehensive business logic, and production-ready deployment configuration. The system provides a solid foundation for fitness centers to manage their class booking operations efficiently.