Fitlife

# Smart Gym Management System

Full-Stack Web Application with Role-Based Access Control

Spring Boot + React.js + MySQL

# Presentation Agenda

- System Overview & Business Value
- Technical Architecture
- Database Design & Schema
- User Roles & Access Control
- Core Features & Functionality
- API Endpoints & Integration
- Security & Authentication
- Deployment & Scalability
- Future Enhancements
- Q&A & Discussion

# System Overview & Business Value

- 🏋️ Full-Stack Gym Management Solution
- 🎯 Comprehensive Role-Based Access Control
- 💼 Streamlined Operations Management
- 📊 Real-Time Analytics & Reporting
- 🔐 Enterprise-Grade Security
- 📱 Responsive Web Application
- 🗄️ Robust Database Architecture
- 🔄 Seamless API Integration

# Technical Architecture

### Frontend Layer

React.js

Bootstrap 5

Responsive Design

### Backend Layer

Spring Boot

Java 17

RESTful APIs

### Database Layer

MySQL 8.0

JPA/Hibernate

Connection Pooling

Secure communication between layers with JWT authentication and HTTPS

# Database Schema Design

### Users

Role-based access

Authentication

### Memberships

Subscription plans

Status tracking

### Equipment

Inventory management

Maintenance tracking

### Gym Classes

Class scheduling

Capacity management

### Training Sessions

Personal training

Booking system

### Payments

Transaction records

Multiple methods

### Class Registrations

Attendance tracking

Enrollment management

# User Roles & Access Control

## ADMIN

Full system access
User management
System configuration

## STAFF

Membership management
Equipment management
Payment processing

## TRAINER

Training sessions
Class teaching
Progress tracking

## MEMBER

Profile management
Session booking
Class registration

# Core Features & Functionality

- 🔐 JWT Authentication & Authorization
- 👥 Multi-Role User Management
- 💳 Membership & Subscription Management
- 🏋️ Equipment Inventory & Maintenance
- 📚 Class Scheduling & Registration
- 👨‍🏫 Personal Training Session Booking
- 💰 Payment Processing & Tracking
- 📊 Real-Time Analytics & Reporting
- 🔍 Advanced Search & Filtering
- 📱 Responsive Web Interface

# API Endpoints & Integration

### Authentication

POST /api/auth/signin

POST /api/auth/signup

### Users

- GET /api/users - Get all users (ADMIN/STAFF)
- GET /api/users/{id} - Get user by ID
- PUT /api/users/{id} - Update user
- DELETE /api/users/{id} - Delete user (ADMIN)

### Memberships

- GET /api/memberships - Get all memberships
- GET /api/memberships/{id} - Get membership by ID
- GET /api/memberships/user/{userId} - Get user's memberships
- POST /api/memberships - Create membership
- PUT /api/memberships/{id} - Update membership
- DELETE /api/memberships/{id} - Delete membership

### Equipment

- GET /api/equipment - Get all equipment
- GET /api/equipment/{id} - Get equipment by ID
- POST /api/equipment - Create equipment
- PUT /api/equipment/{id} - Update equipment
- DELETE /api/equipment/{id} - Delete equipment

### Classes

- GET /api/gym-classes - Get all classes
- POST /api/gym-classes - Create class
- PUT /api/gym-classes/{id} - Update class
- DELETE /api/gym-classes/{id} - Delete class

### Sessions

- GET /api/training-sessions - Get all sessions
- POST /api/training-sessions/book - Book session (MEMBER)
- POST /api/training-sessions - Create session
- PUT /api/training-sessions/{id} - Update session
- DELETE /api/training-sessions/{id} - Delete session

### Class Registrations

- GET /api/class-registrations - Get all registrations
- POST /api/class-registrations/register - Register for class
- PUT /api/class-registrations/{id}/status - Update registration status

### Payments

- GET /api/payments - Get all payments
- POST /api/payments - Create payment
- PUT /api/payments/{id}/status - Update payment status

# Security & Authentication

### JWT Tokens

Secure token-based authentication

### Spring Security

Role-based access control

### CORS

Cross-origin resource sharing

### Input Validation

Data sanitization & validation

### HTTPS

Secure communication protocols

# Deployment & Scalability

- 🚀 Spring Boot Application Server
- 🗄️ MySQL Database with Connection Pooling
- 🌐 React.js Frontend with Bootstrap
- 🔧 Maven Build & Dependency Management
- 📦 Docker Containerization Ready
- ☁️ Cloud Deployment Compatible
- 📊 Monitoring & Logging Integration
- 🔄 CI/CD Pipeline Support

# Future Enhancements & Roadmap

### Mobile App

iOS & Android
applications

### AI Integration

Smart scheduling
& recommendations

### Payment Gateway

Stripe, PayPal
integration

### Analytics

Advanced reporting
& insights

### Notifications

Email & SMS
alerts

### IoT Integration

Smart equipment
monitoring

# Questions & Discussion

- ❓ Technical Questions
- 💡 Feature Requests
- 🔧 Implementation Details
- 📊 Performance Considerations
- 🚀 Deployment Strategies
- 🔄 Integration Options
- 📱 User Experience
- 🔐 Security Concerns

# Thank You!

🎯

## Gym Management System

# Security & Authentication

- JWT Tokens – Secure token-based authentication
- BCrypt – Password encryption & hashing
- Spring Security – Role-based access control
- CORS – Cross-origin resource sharing
- Input Validation – Data sanitization & validation
- HTTPS – Secure communication protocols

# Deployment & Scalability

- □□Spring Boot Application Server
- MySQL Database with Connection Pooling
- □□React.js Frontend with Bootstrap
- □□Maven Build & Dependency Management
- □□Docker Containerization Ready
- Cloud Deployment Compatible
- □□Monitoring & Logging Integration
- □□CI/CD Pipeline Support

# Future Enhancements & Roadmap

- Mobile App – iOS & Android applications
- AI Integration – Smart scheduling & recommendations
- Payment Gateway – Stripe, PayPal integration
- Analytics – Advanced reporting & insights
- Notifications – Email & SMS alerts
- IoT Integration – Smart equipment monitoring

# Questions & Discussion

- ? Technical Questions
- □ □ Feature Requests
- □ □ Implementation Details
- □ □ Performance Considerations
- □ □ Deployment Strategies
- □ □ Integration Options
- □ □ User Experience
- □ □ Security Concerns

# Thank You!

- Gym Management System
- □ □Contact: development@example.com
- □ □Website: www.example.com
- □ □Phone: +1 (555) 123-4567

- Ready for Implementation! – The system is fully functional and ready for deployment
- Any additional questions?