

# YogiTrack Studio Management System

Part 1 Project Report

Robert Hamilton

September 21, 2025

**GitHub Repository:** <https://github.com/rghamilton3/yogitrack-prototype>

**Live Application:** <https://yogitrack-prototype-a1daf9b97aac.herokuapp.com/>

# 1 Project Overview

YogiTrack is a MERN stack application designed for yoga studio management. The system enables administrators to manage instructors, customers, and classes with automated conflict detection and notification systems.

## 2 Implemented Use Cases

### 2.1 Use Case 1: Add Instructor

Allows adding new yoga instructors with duplicate name validation and automatic confirmation messaging to management.

### 2.2 Use Case 2: Add Class

Enables class creation with intelligent schedule conflict detection, alternative time slot suggestions, and instructor notifications.

### 2.3 Use Case 4: Add Customer

Facilitates customer registration with senior status tracking and comprehensive profile management.

## 3 Architecture & Design Decisions

### 3.1 Technology Stack

- **Frontend:** React 19 with functional components and hooks
- **Backend:** Express.js 5 with RESTful API design
- **Database:** MongoDB with Mongoose ODM
- **Build Tool:** Webpack with Babel transpilation
- **Deployment:** Heroku with GitHub Actions CI/CD

### 3.2 Key Design Patterns

- Reusable EntityForm component eliminating code duplication
- Centralized API service layer with error handling
- Embedded document schemas for complex schedule management
- Conflict detection algorithms with alternative suggestions

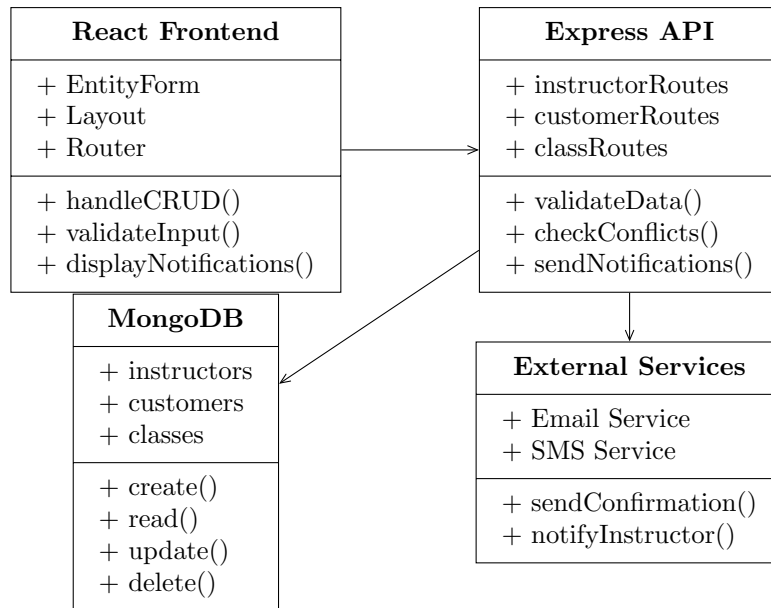


Figure 1: High-Level System Architecture

## 4 System Architecture

## 5 Database Schema

## 6 Key Implementation Features

### 6.1 Schedule Conflict Detection

Implemented intelligent algorithm that checks for time overlaps and provides alternative scheduling suggestions when conflicts are detected.

### 6.2 Automated Notifications

Integrated confirmation messaging system that notifies managers and instructors upon successful operations.

### 6.3 Responsive UI Components

Created reusable React components with consistent styling and error handling across all entity management interfaces.

### 6.4 CI/CD Pipeline

Established automated deployment workflow using GitHub Actions with Heroku integration for continuous delivery.

## 7 Conclusion

The YogiTrack system successfully implements core studio management functionality with robust conflict detection, automated notifications, and a scalable MERN architecture. The reusable component design and comprehensive API layer provide a solid foundation for future feature expansion.

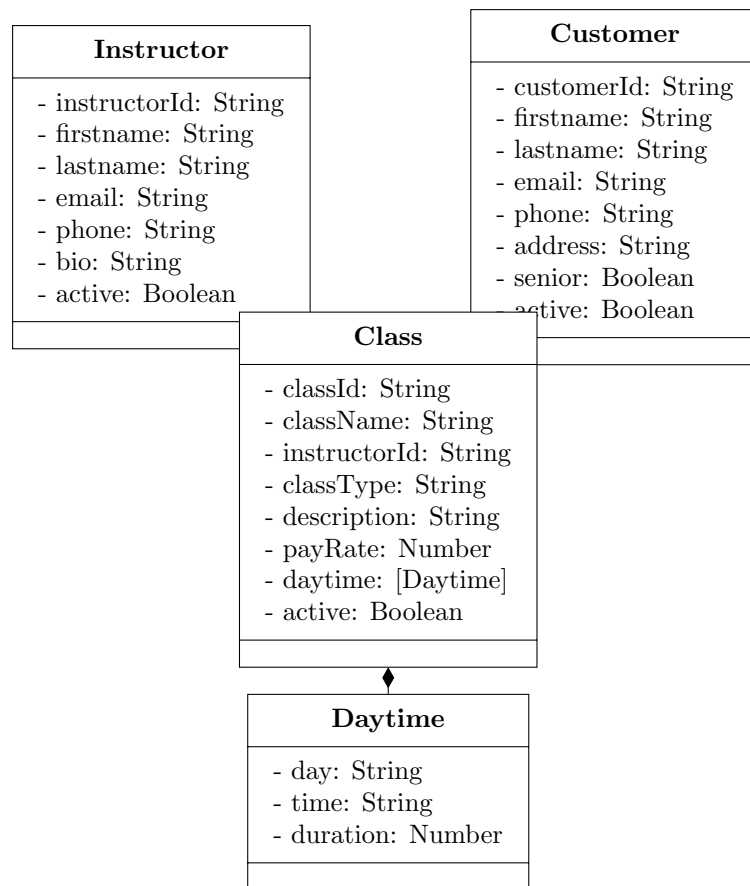


Figure 2: Database Entity Relationship Diagram