Personal Fitness Management System

# Team Members

Team ID: NM2025TMID29248

Team Size:04

Team Leader: Sivanandhini S

Team Member: Gopika M

Team Member: Navanitha S

Team Member: Yogeswari N

# Project Overview

Purpose:  
The Personal Fitness Management System helps users manage their fitness journey by tracking exercises, workouts, diet plans, and progress. The project provides a simple interface to log fitness data and visualize progress.

Features:  
- Add, edit, and delete workout records  
- Track daily/weekly progress  
- View dashboard with analytics  
- Manage personal fitness goals  
- Backend API with FastAPI for data handling  
- Frontend with React (Vite) for user interface

# Architecture

Frontend (React + Vite):  
- Components organized into components/ and pages/  
- React Router for navigation between pages  
- State handled via React hooks (useState, useEffect) and Context API

Backend (FastAPI):  
- REST APIs for managing workouts and progress  
- Data stored in database (SQLite/PostgreSQL – configurable)  
- CORS enabled for frontend-backend communication

# Setup Instructions

Prerequisites:  
- Node.js (>= 16)  
- Python (>= 3.9)  
- Git  
- pip / virtualenv

Installation:  
1. Clone repository:  
 git clone https://github.com/sivanandhini4466-create/personal-fitness-management-system.git  
 cd personal-fitness-management-system

Backend (FastAPI):  
 cd backend  
 python -m venv venv  
 source venv/bin/activate # (Windows: venv\Scripts\activate)  
 pip install -r requirements.txt  
 uvicorn main:app --reload  
 Backend runs on http://127.0.0.1:8000

Frontend (React):  
 cd frontend  
 npm install  
 npm run dev  
 Frontend runs on http://localhost:5173

# Folder Structure

personal-fitness-management-system/  
│── backend/ # FastAPI backend  
│ ├── main.py # API entry point  
│ ├── requirements.txt  
│  
│── frontend/ # React (Vite) frontend  
│ ├── src/  
│ │ ├── components/ # Reusable React components  
│ │ ├── pages/ # Pages (Dashboard, Workouts, etc.)  
│ │ ├── assets/ # Images, CSS  
│ ├── package.json  
│  
│── README.md

# Running the Application

1. Start backend first (uvicorn main:app --reload)  
2. Start frontend (npm run dev)  
3. Access app via browser: http://localhost:5173

# Component Documentation

- DashboardPage – Displays progress overview & stats  
- WorkoutForm – Add/edit workouts  
- WorkoutList – View logged workouts  
- Navbar – Navigation across app

# State Management

Global State: Context API used for authentication & user data  
Local State: useState hooks for forms and filters

# User Interface

Responsive layout (mobile + desktop)  
Clean dashboard UI for quick tracking  
Forms for easy workout input

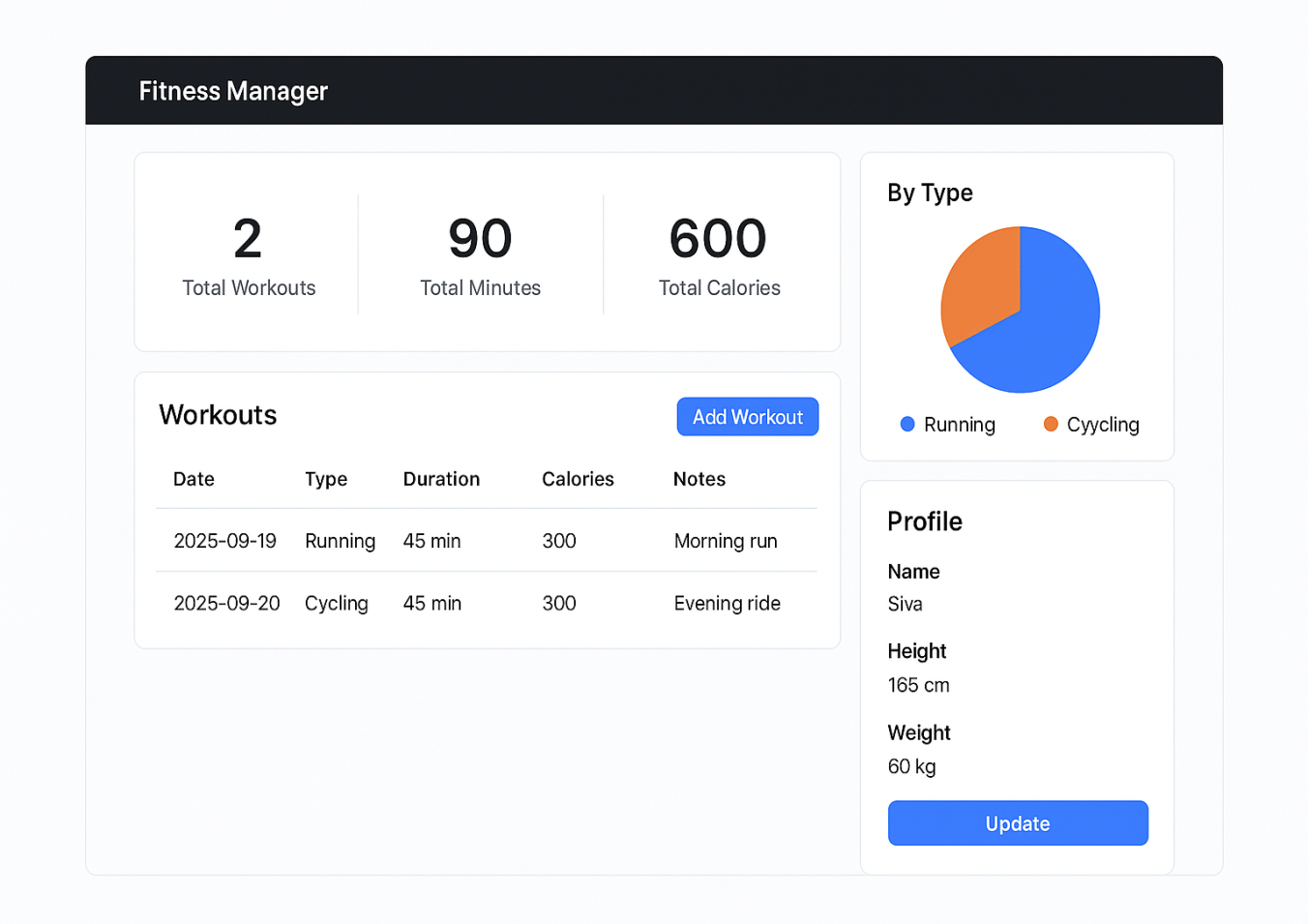
# Styling

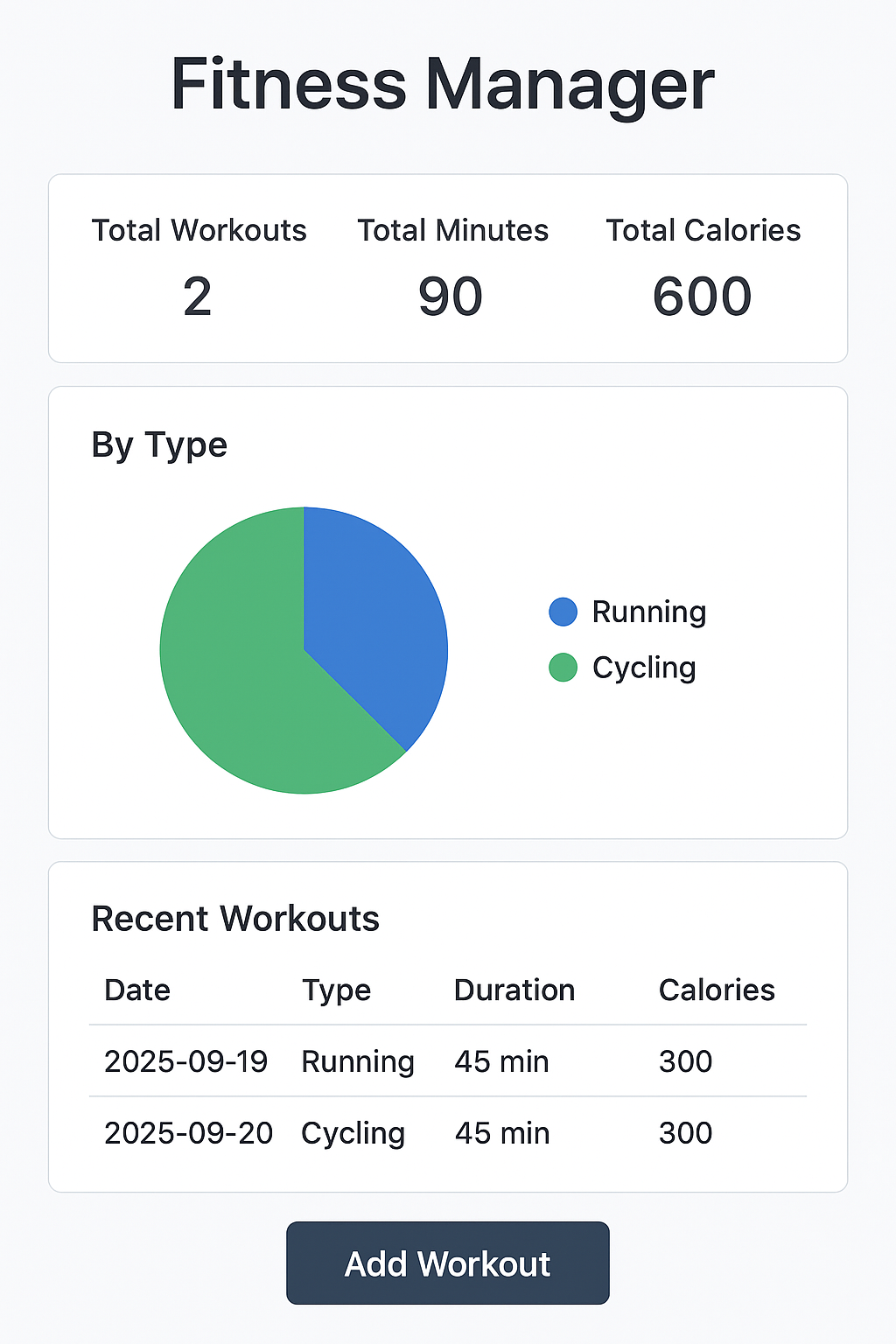
Tailwind CSS used for styling  
Dark/light mode possible (future enhancement)

# Testing

Frontend: React Testing Library + Jest  
Backend: Pytest for API endpoints  
Coverage via npm test (frontend) & pytest --cov (backend)

# Screenshots or Demo





# Known Issues

Authentication not fully implemented  
Limited analytics (basic charts only)

# Future Enhancements

User authentication (JWT-based login)  
Advanced analytics & reports  
Integration with wearable devices (Fitbit, Apple Watch)  
Cloud deployment (Heroku/Vercel)