CC Jackfruit Problem Phase - 1

Name: Achyuth N G   **SRN:** PES1UG22AM008
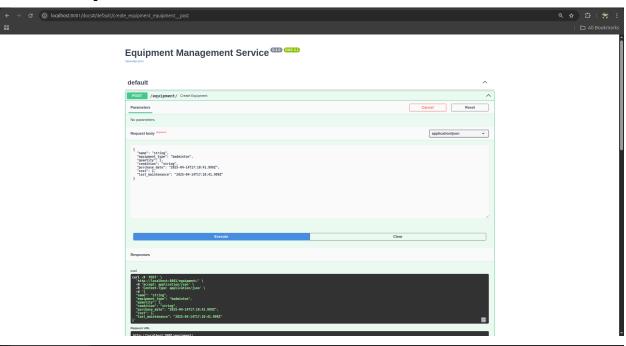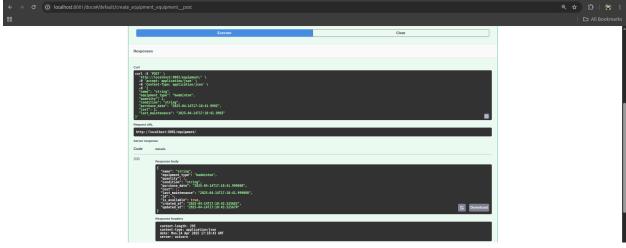   Akshaj B Seerpu   PES1UG22AM018
          Arpan Rishon Mascarenhas   PES1UG22AM031
          Channabasappa V Patil   PES1UG22AM048

Sec: 6A CSE(CSE & ML)

_____

1.Output screenshot:

Tech Stack:

- **Containerization**:

  - Docker

  - Docker Compose

- **Database**:

  - PostgreSQL 15

- **Network**:

  - Docker Bridge Network (`virtual-labs-network`)

Backend Framework & Libraries

- **Core Framework**:

  - FastAPI (v0.68.0+): Modern Python web framework

  - Uvicorn (v0.15.0+): ASGI web server implementation

- **Database Tools**:

  - SQLAlchemy (v1.4.0+): SQL toolkit and ORM

  - Psycopg2 (v2.9.1+): PostgreSQL adapter for Python

  - Alembic (v1.7.0+): Database migration tool

- **Data Validation & Schema**:

  - Pydantic (v1.8.0+): Data validation using Python type annotations

  - Email-Validator (v2.0.0+): Email validation library

Development & Testing

- **Testing**:

  - pytest (v6.2.4+): Testing framework

  - httpx (v0.18.2+): HTTP client for testing

Service-Specific Components

1. **Staff Service** (Port 8004)

2. **Equipment Service** (Port 8001)

3. **Gym Subscription Service** (Port 8002)

4. **Sports Coaching Service** (Port 8003)

Shared Components

- **Environment Configuration**:
    - Python-dotenv for environment variable management
    - Service-specific environment variables
    - Shared database configuration

- **API Documentation**:
    - Swagger UI (`/docs`)
    - ReDoc (`/redoc`)
    - OpenAPI specification

File Structure:

```
sports_management/
├── shared/                    # Shared utilities and configurations
├── service_name/              # For each service
│   ├── app/
│   │   ├── models.py          # Database models
│   │   ├── schemas.py         # Pydantic schemas
│   │   ├── main.py            # FastAPI application
│   │   └── database.py        # Database configuration
│   ├── Dockerfile             # Container configuration
│   └── requirements.txt       # Service dependencies
└── docker-compose.yml         # Multi-container orchestration
```