
# INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT AKURDI, PUNE

Documentation On

## "Fitness Club-Gym Services"

PG-DAC SEPTEMBER 2022

Submitted By:

**Group No: 72**

- **Shubham Gurav** **220941220061**
- **Pushpdant Patil** **220941220125**

**Mr. Rohit Puranik**
**Centre Coordinator**

**Mrs. Manjiri Deshpande**
**Project Guide**

# Table of Contents

# List of Figures

# 1.INTRODUCTION

## Introduction:

This document communicates the business requirements and scope for developing Gym Service for a company. The scope of this document is to define the functional and non- functional requirements, business rules and other constraints requirements.
Now a day's online service is the best competitive edge for any organization. Our fitness management website provides best platform for ease of access to the gym managers, trainers and also for customers. User can check his updates online anytime about his/her fitness, diet plan etc. There is a need for online healthcare maintenance online. This project provides user friendly customer and trainer interaction.

## Problem Statement:

Existing Gyms works without any website for providing services to their customers. Managers have to keep records on papers and registers. There is no any way for trainers and customers to manage their workout progress. Customers need to visit gym for checking facilities and packages provided by different gyms.
Fitness club-Gym Services is intended to provide complete solution for Gym owner, trainers & customers through a single gateway using internet. It allows owners to manage their gym, customers to view packages provided by gym ,search and choose trainers and get information about gym equipments and purchase the convenient package to get membership. It allows gym trainers to evaluate workout and diet report of gym members. Gym managers can manage their daily gym schedule and send notifications to customers about same. The administrator module will be able to manage branch activity, trainer activity and payments.

## Aims and Objective:

This product aimed toward a person who don't want to visit the gym to see functionalities and packages provided by that gym to get membership, he/she can use the web application for ease.

In other words, our Gym Management portal has, following objectives:

- Simple database is maintained.
- Easy operations for the user and the admin of the system.
- User interfaces are user accommodating and attractive; it takes very less time for the operator to use the system.
- This system will provide complete solution for Gym owners to take their business online.

# 2. OVERALL DESCRIPTION

**Proposed Methodology:**

This system brings ease in the communication and business of B2C field. It provides the complete functionality to owner This system allows gym managers to manage users and full application, manage gym shifts and the members to search gyms, apply for membership and view workouts while it allows trainer to create schedule, diet chart.

This product aimed toward a person who don't want to visit the gym to see functionalities and packages provided by that gym to get membership, he/she can use the web application for ease.

Fitness Management Website provides better users health and diet plan and workout plan maintaining their health care and taking care of all their health information.

Our System provides a very user-friendly platform where Member can easily search the gym trainer and check his updates online anytime about his/her fitness, diet plan etc.

Our system is aimed efficient management of various tasks like Generating diet and workout plan for members on weekly basis, managing fees payment. Digitally monitor daily activities along with managing all the resources and information on a single platform.

## OPERATING ENVIRONMENT:

### Server Side:

**Processor:** Intel® Xeon® processor 3500 series

**HDD:** Minimum 500GB Disk Space

**RAM:** Minimum 4GB

**OS:** Windows 10

**Database:** MySQL

### Client Side (minimum requirement):

**Processor:** Intel Dual Core

**HDD:** Minimum 80GB Disk Space

**RAM:** Minimum 4GB

**OS:** Windows 7 or above

### Design and Implementation Constraints:

- The application will use ReactJS, Axios and CSS as main web technologies.
- HTTP protocol is used as communication protocol. FTP is used to upload the web application in live domain and the client can access it via HTTP protocol.
- SMTP protocol is used for Email communication
- Several types of validations make this web application a secured one and SQL Injections can also be prevented.
- Since Fitness Club is a web-based application, internet connection must be established.

# 3. REQUIREMENTS SPECIFICATION.

## External Interface Requirements:

User Interfaces:

- All the users will see the same page when they enter in this website. This page asks the users a username and a password.
- After being authenticated by correct username and password, user will be redirect to their corresponding profile where they can do various activities.
- The user interface will be simple and consistence, using terminology commonly understood by intended users of the system. The system will have simple interface, consistence with standard interface, to eliminate need for user training of infrequent users.

Hardware Interfaces:

- No extra hardware interfaces are needed.
- The system will use the standard hardware and data communication resources.

This includes, but not limited to, general network connection at the server/hosting site, network server and network management tools.

Application Interfaces:

**Web Browser:**

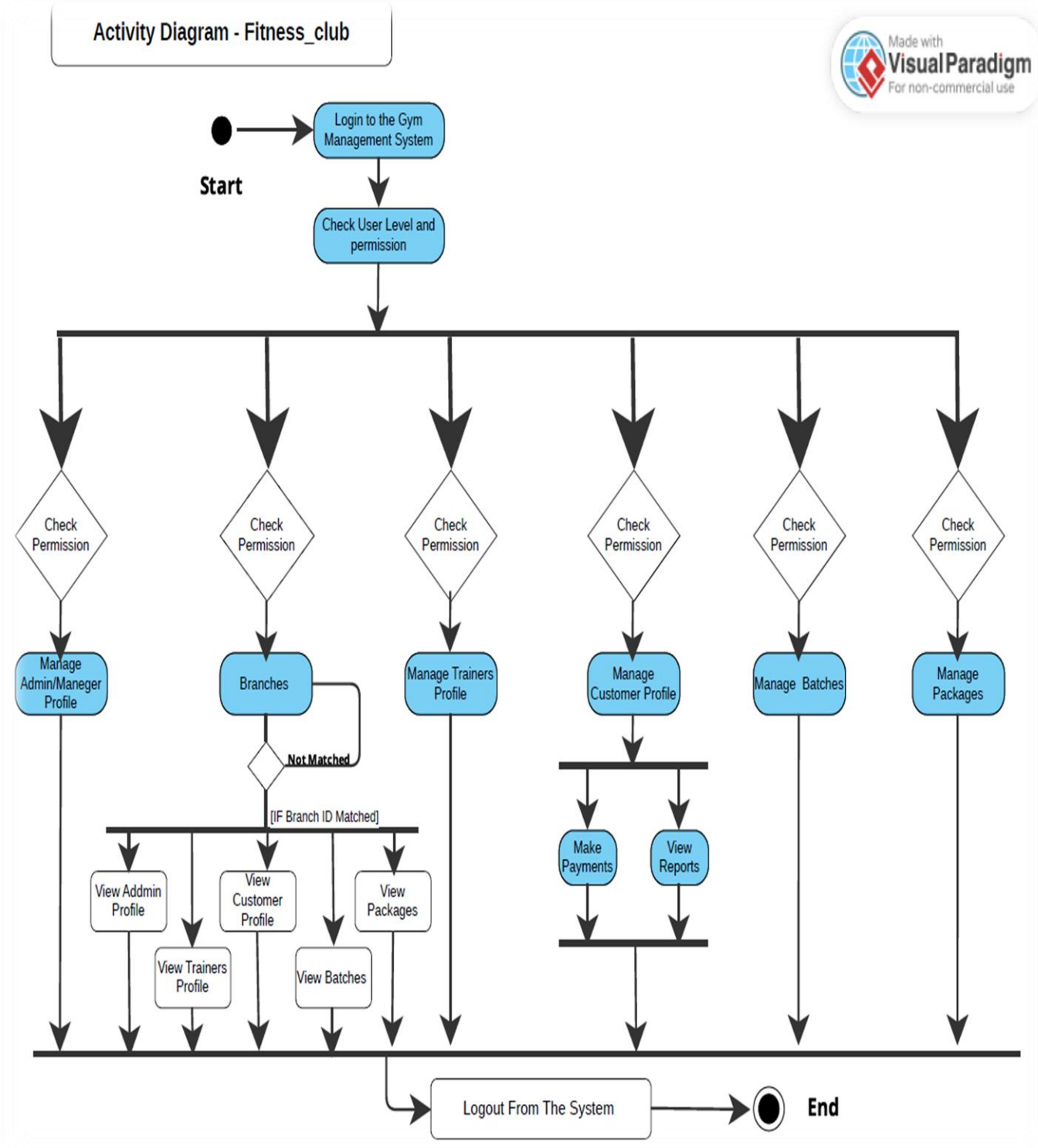
The system is a web-based application; clients need a modern web browser such as Mozilla Firebox, Internet Explorer, Opera, and Chrome. The computer must have an Internet connection in order to be able to access the system.

Communications Interfaces:

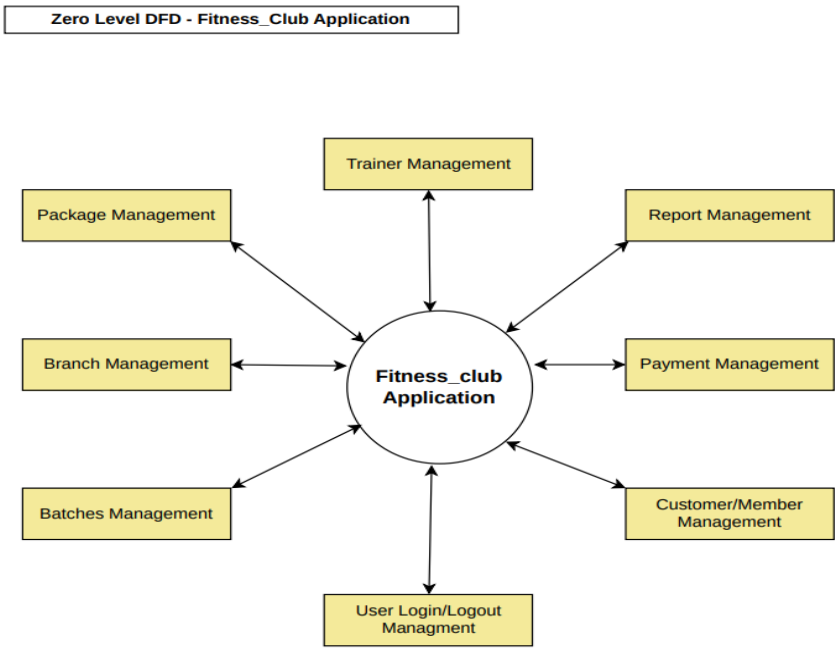
- This system uses communication resources which includes but not limited to, HTTP protocol for communication with the web browser and web server and TCP/IP network protocol with HTTP protocol.
- This application will communicate with the database that holds all the booking information. Users can contact with server side through HTTP protocol by means of a function that is called HTTP Service. This function allows the application to use the data retrieved by server to fulfil the request fired by the us.

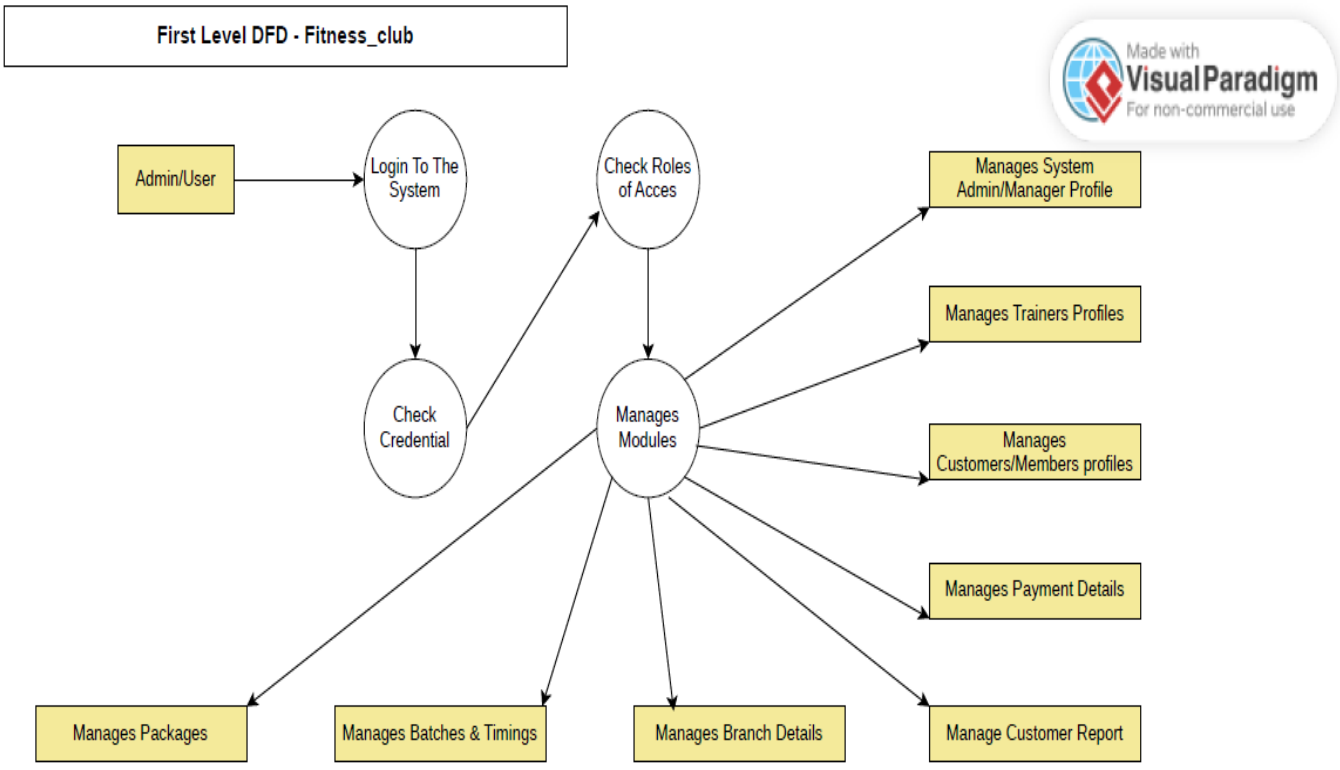
# 4. SYSTEM DIAGRAMS

- **Activity Digram**

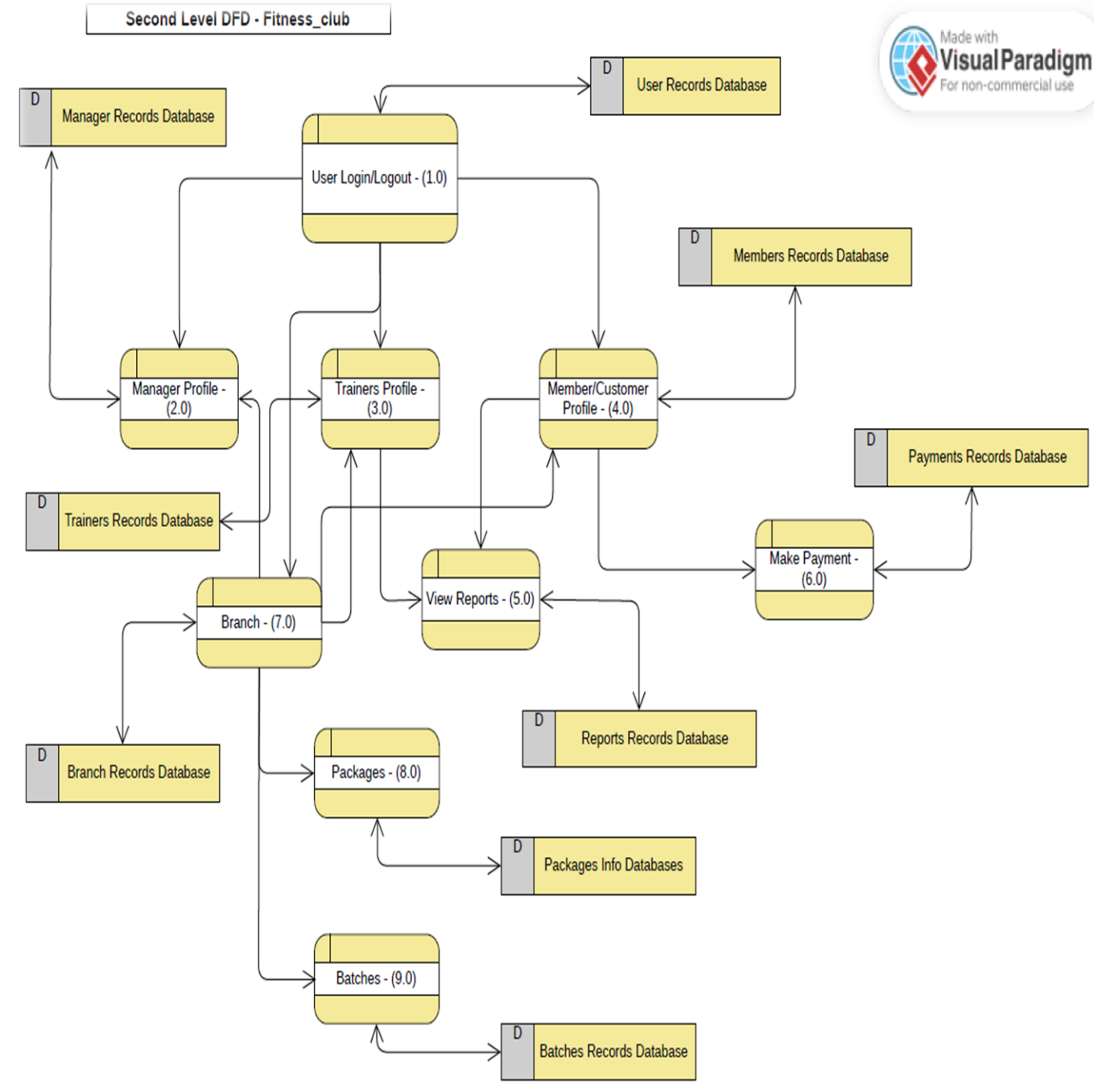


- **Data Flow Diagrams**

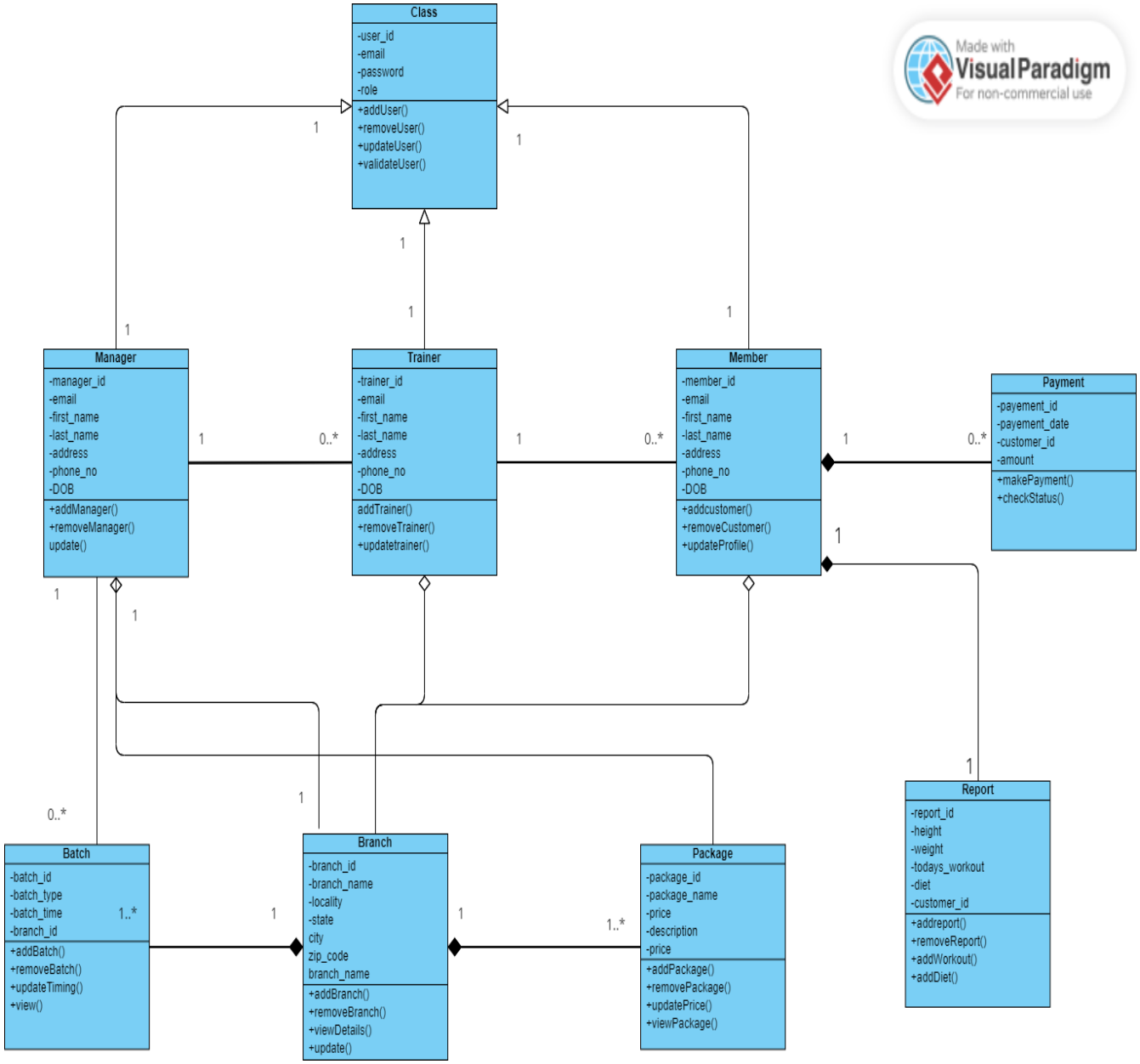


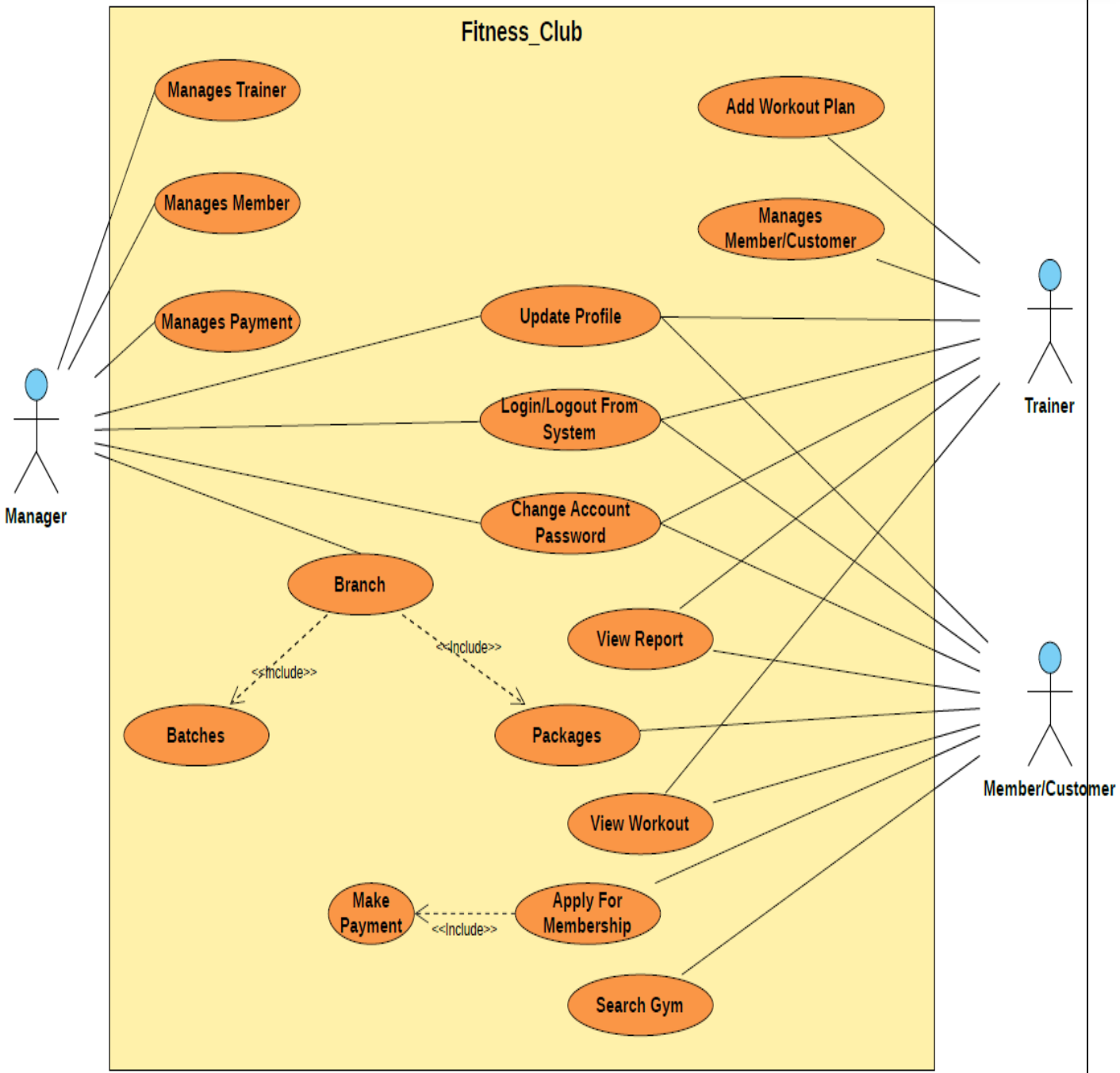
- **Second level DFD**



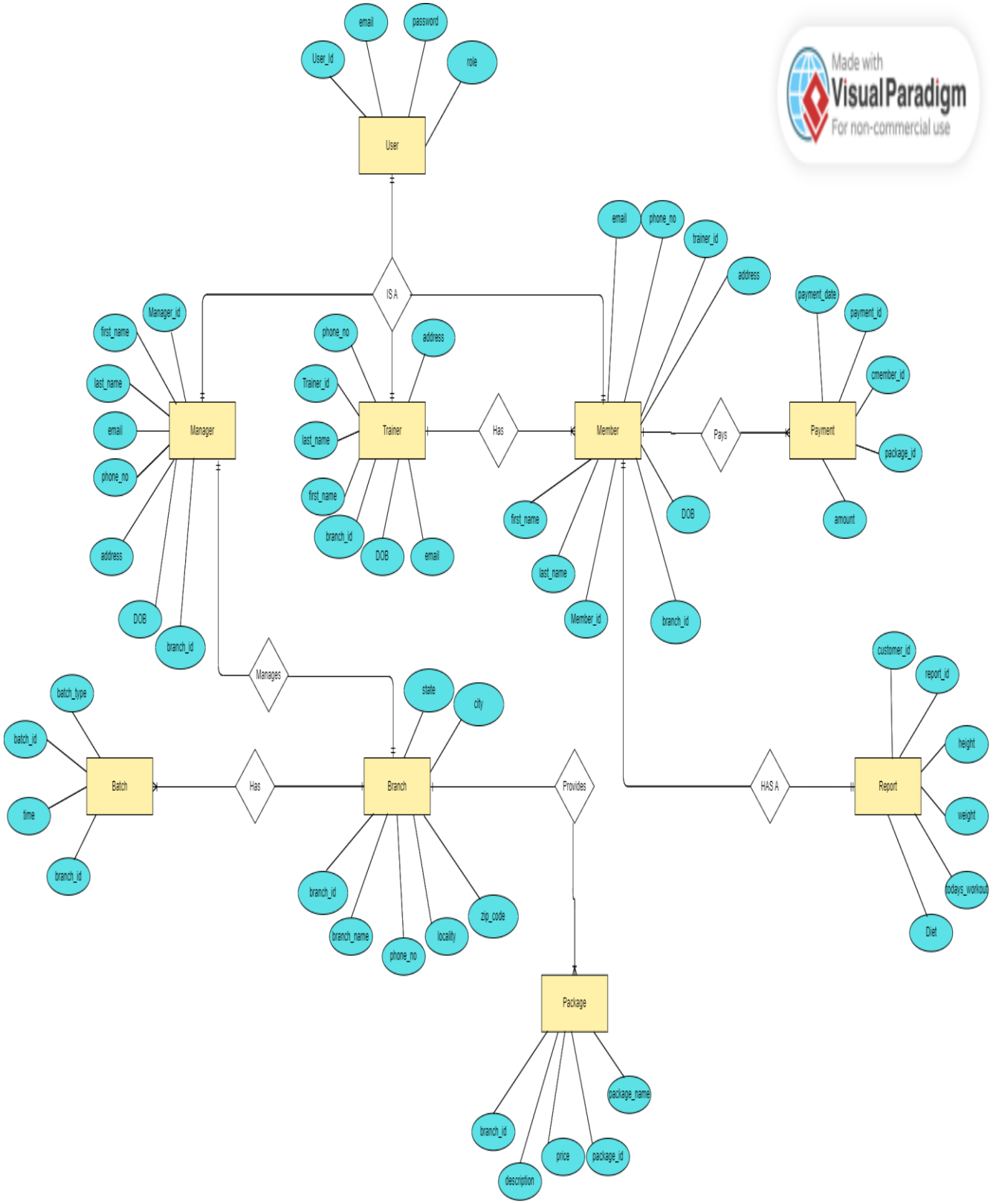
- **Class Diagram**



- **Use Case Diagram**



- **ER Diagram**


email
password
User_Id
role
User
Made with VisualParadigm For non-commercial use
ISA
email
phone_no
trainer_id
address
payment_date
payment_id
Manager_id
first_name
last_name
email
phone_no
address
DOB
branch_id
Manager
phone_no
address
Trainer_id
last_name
first_name
branch_id
DOB
email
Trainer
Has
Member
Pays
Payment
cmember_id
package_id
amount
first_name
last_name
Member_id
DOB
branch_id
Manages
customer_id
report_id
batch_type
batch_id
time
branch_id
Batch
Has
state
city
Branch
Provides
HAS A
Report
height
weight
todays_workout
Diet
branch_id
branch_name
phone_no
locality
zip_code
Package
package_name
branch_id
price
package_id
description

# 5. TABLE STRUCTURE

**User:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| UserId | bigint | NO | PRI | NULL | auto_increment |
| email | Varchar(255) | NO | | NULL | |
| password | Varchar(255) | NO | | NULL | |
| role | Varchar(255) | YES | | NULL | |

**Manager:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| ManagerId | bigint | NO | PRI | NULL | auto_increment |
| address | Varchar(100) | YES | | NULL | |
| Dob | date | YES | | NULL | |
| email | Varchar(30) | NO | UNI | NULL | |
| first_name | Varchar(20) | YES | | NULL | |
| last_name | Varchar(20) | YES | | NULL | |
| Password | Varchar(255) | NO | | NULL | |
| phone_no | double | YES | | NULL | |
| role | Varchar(255) | YES | | NULL | |
| branch_id | bigint | YES | MUL | NULL | |
| user_id | bigint | YES | MUL | NULL | |

**Batch:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| BatchId | bigint | NO | PRI | NULL | auto_increment |
| batch_time | time | YES | | NULL | |
| batch_type | Varchar(250) | YES | | NULL | |
| branch | bigint | YES | | NULL | |

**Trainer:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| TrainerId | bigint | NO | PRI | NULL | auto_increment |
| address | Varchar(100) | YES | | NULL | |
| Dob | date | YES | | NULL | |
| email | Varchar(30) | NO | UNI | NULL | |
| first_name | Varchar(20) | YES | | NULL | |
| last_name | Varchar(20) | YES | | NULL | |
| Password | Varchar(255) | NO | | NULL | |
| phone_no | double | YES | | NULL | |
| role | Varchar(255) | YES | | NULL | |
| branch_id | bigint | YES | MUL | NULL | |
| user_id | bigint | YES | MUL | NULL | |

**Member:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| MemberId | bigint | NO | PRI | NULL | auto_increment |
| address | Varchar(100) | YES | | NULL | |
| Dob | date | YES | | NULL | |
| email | Varchar(30) | NO | UNI | NULL | |
| first_name | Varchar(20) | YES | | NULL | |
| last_name | Varchar(20) | YES | | NULL | |
| Password | Varchar(255) | NO | | NULL | |
| phone_no | double | YES | | NULL | |
| role | Varchar(255) | YES | | NULL | |
| branch_id | bigint | YES | MUL | NULL | |
| trainer_id | bigint | YES | MUL | NULL | |
| user_id | bigint | YES | MUL | NULL | |

**Branch:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| BranchId | bigint | NO | PRI | NULL | auto_increment |
| state | varchar(255) | YES | | NULL | |
| branch_name | varchar(255)) | NO | | NULL | |
| city | varchar(255) | YES | | NULL | |
| locality | varchar(255) | YES | | NULL | |
| phone_no | double | YES | | NULL | |
| zip_code | varchar(255) | YES | | NULL | |

**Packages:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| PackageId | bigint | NO | PRI | NULL | auto_increment |
| description | varchar(100) | YES | | NULL | |
| package_name | varchar(20) | YES | | NULL | |
| package_price | double | YES | | NULL | |
| branch | bigint | YES | MUL | NULL | |

**Payment:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| PaymentId | bigint | NO | PRI | NULL | auto_increment |
| amount | double | YES | | NULL | |
| date | datetime(6) | YES | | NULL | |
| member | bigint | YES | MUL | NULL | |
| package | bigint | YES | MUL | NULL | |

**Report:**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| ReportId | bigint | NO | PRI | NULL | auto_increment |
| diet | varchar(1000) | YES | | NULL | |
| height | double | NO | | NULL | |
| weight | double | NO | | NULL | |
| todays_workout | varchar(1000) | YES | | NULL | |
| member_id | bigint | YES | MUL | NULL | |

# 6. CONCLUSION

This system brings ease in the communication and business of B2C field. It provides the complete functionality to owner This system allows admin to manage users and full application, manage gym shifts and the members to search gyms, apply for membership and view workouts while it allows trainer to create schedule, diet chart and add workout plans.

This system provides opportunity to Gym owners to expand their business online. Saves time and efforts of customers to right gym and reduces overall paper work of managing records and registers. Customers and Trainers can receive notifications via email.

- **Future Scope:**

This project can be enhanced further by adding payment gateway to reduce the maintenance of cash for Membership purchase payments. Online Workout tutorials and online Expert sessions can be hosted on this site for better customer satisfaction. The software is flexible enough to be modified and implemented as per future requirements. We have tried our best to present this free and user–friendly website to Institutes.

# 6.REFERENCES

- **References:**

  - React – A JavaScript library for building user interfaces (reactjs.org)
  - Bootstrap · The most popular HTML, CSS, and JS library in the world. (getbootstrap.com)
  - React Tutorial (w3schools.com)
  - Learn Spring Boot | Baeldung
  - Spring Data JPA - Reference Documentation
  - **cult.fit - Bring gym home**
  - **Gym Management-UML**