Design

Document

Online Gym Management System .

**Index**

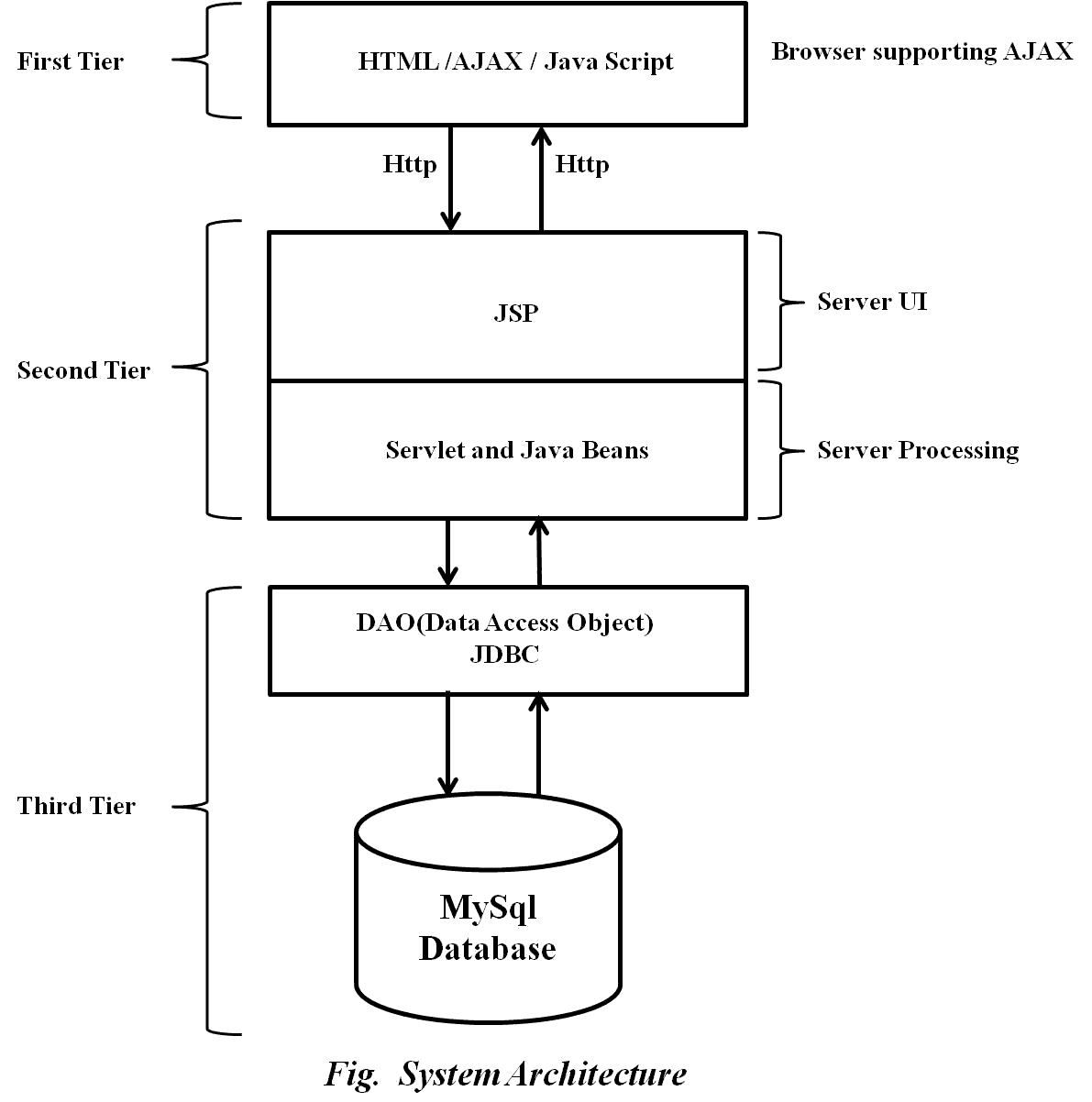
|  |  |  |
| --- | --- | --- |
| Sr No | Content | Page No |
| 1 | Introduction | 3 |
| 2 | Architectural Design | 3 |
| 3 | High Level Design | 6 |
|  | 3.1 E-R Diagram | 6 |
|  | 3.2 Page Navigation Diagram | 7 |
|  | 3.3 Data Flow Diagram | 9 |
|  | 3.4 Deployment Diagram | 12 |
| 4 | Low Level Design | 13 |
|  | 4.1 Database Design | 13 |
|  | 4.2 Stored Procedure | 16 |
|  | 4.3 Details Of Page Navigation | 17 |

* **Introduction:**

This document is meant for the description of the structure and the database which we are using in this project. This document gives brief description about Architecture of the system, E-R diagram of the system and the table descriptions, the page navigation diagrams and the detail description for the page navigation.

* **Architecture Design:**

Following diagram shows the details of the Online Gym Management System architecture.



This System consist of three tiers as listed below,

* First tier
* Second tier
* Third tier

**First Tier:**

This tier is used for user interface and it is called as client tier. In this tier we are using AJAX because of it provides better interactivity, easier navigation, compact. The use of java script facilities us for the client side validation. That’s why in first tier we are using the java script. We are using HTML for the presentation purpose.

**Second Tier:**

Second Tier is comprises of two parts listed below,

* **Server UI**

In this part of second tier we are using JSP, because it provides better UI to system, as well as it provides the dynamically designing of pages.

* **Server Process**

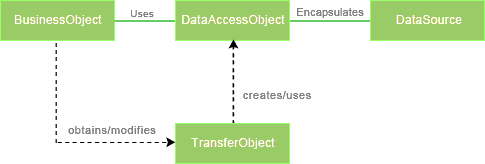
Servlet API is standard and freely available on the internet (like JSPs) servlets have the advantages like ease of development & platform independence (like Java) they can access all the J2SE and J2EE APIs can take the full advantage & capabilities of the Java programming language.

**Third Tier:**

Third tier consist of a Data Access Object (DAO) and the back end i.e. the database of e-farming system.

**Data Access Object (DAO):**

Data access object layer has proven good in separate business logic layer and persistent layer. The DAO design pattern completely hides the data access implementation from its clients.

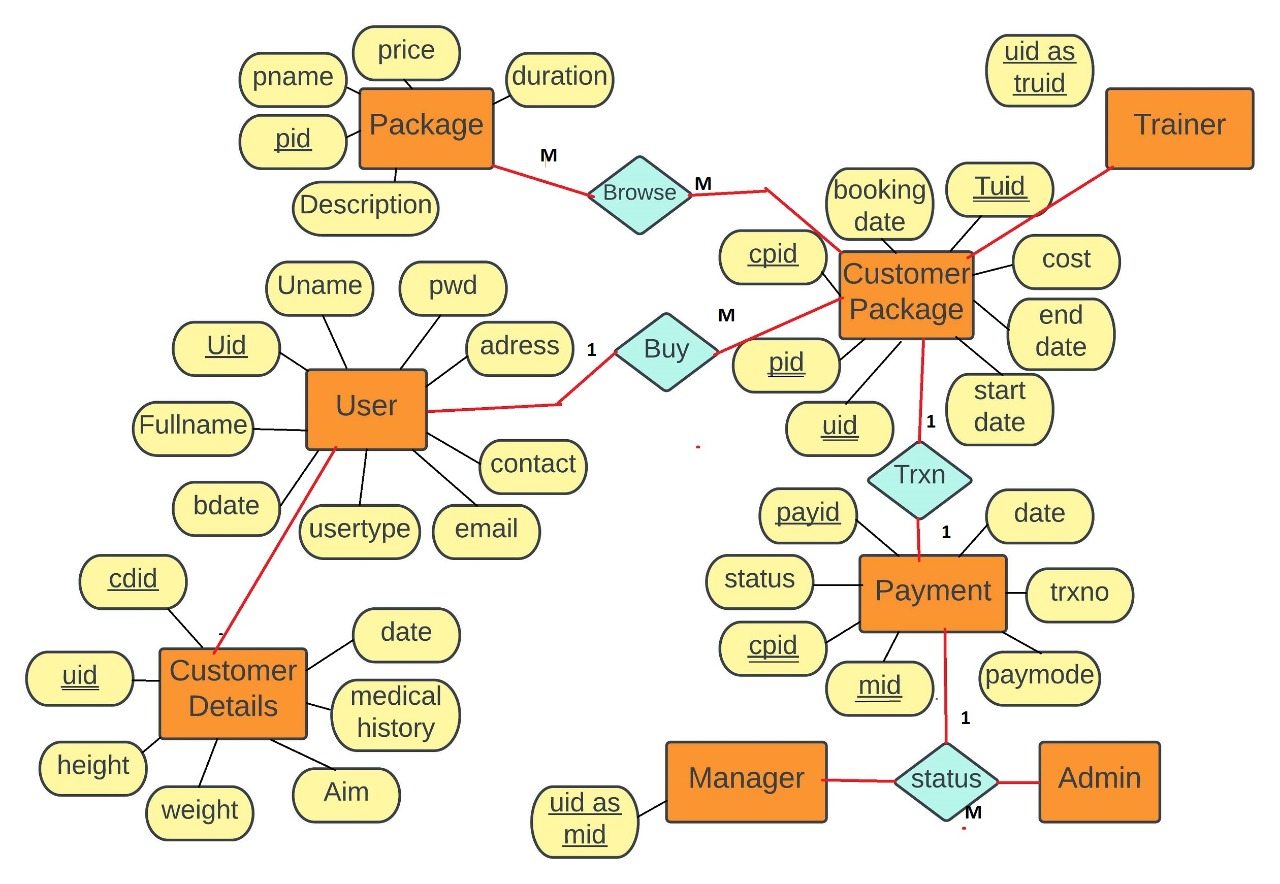


*Fig Data Access Object (DAO) Mechanism*

**Java Database Connectivity (JDBC):**

JDBC is used to provide database connectivity from java to database. Using Java database connectivity we can update/retrieve data to/from database with java programs. The main advantage of using JDBC is we can execute database queries by the program so that we can utilize the functionality provided by the database (with the queries). More over we can use triggers too. JDBC provides much other functionality (like the functions provided by CallableStatemtent class) to manage the data. Additionally, loading the driver will be different to different databases.

* **High Level Design:**
* **E-R Diagram:**



Above E-R Diagram shows that database of Gym Management system consist of following entities:

* **User**

This entity contains the Userid, Username, Password, firstname, lastname, Email , ContactNo , Address, UserType, Birthdate

* **Package**

This entity contains the Packageid, Packagename, Description, Duration, Price

* **CustomerPackage**

This entity contains the CustPackageid, Packageid, Userid, BookingDateid, StartDate, EndDate, Cost

* **Payment**

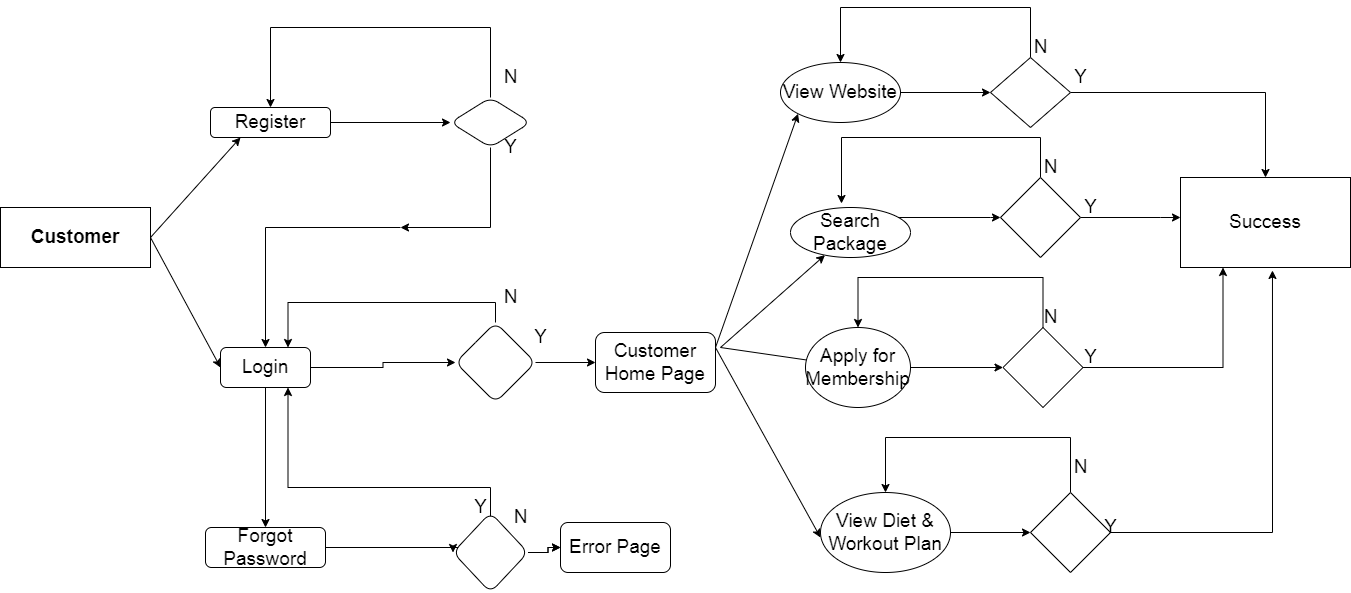
This entity contains the Aid, Paymentid, Status, Custpackageid, Transaction mode, Date, ManagerId.

* **CustomerDetails**

This entity contains the CustomerDetailsid, Userid, Date, Height, Weight, Medical History, Aim

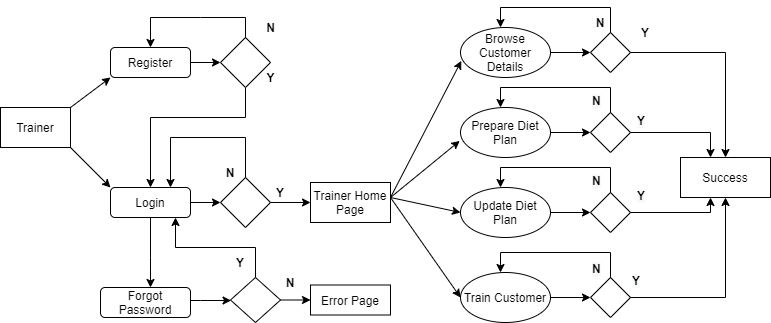
* **Page Navigation Diagram:**
* **Customer**

Following diagram explains the page navigation for the customer module:



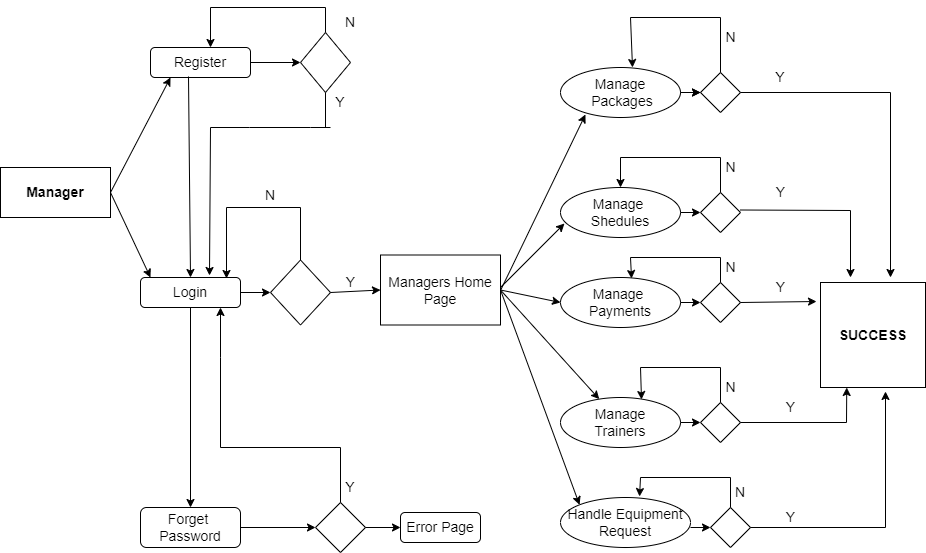
* **Trainer**

Following diagram explains the page navigation for the Trainer module:



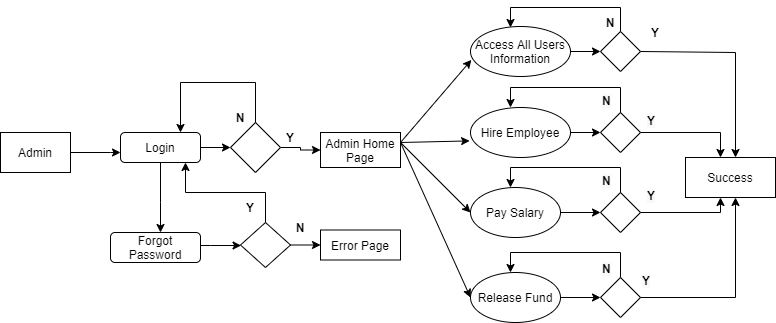
* **Manager**

Following diagram explains the page navigation for the Manager module

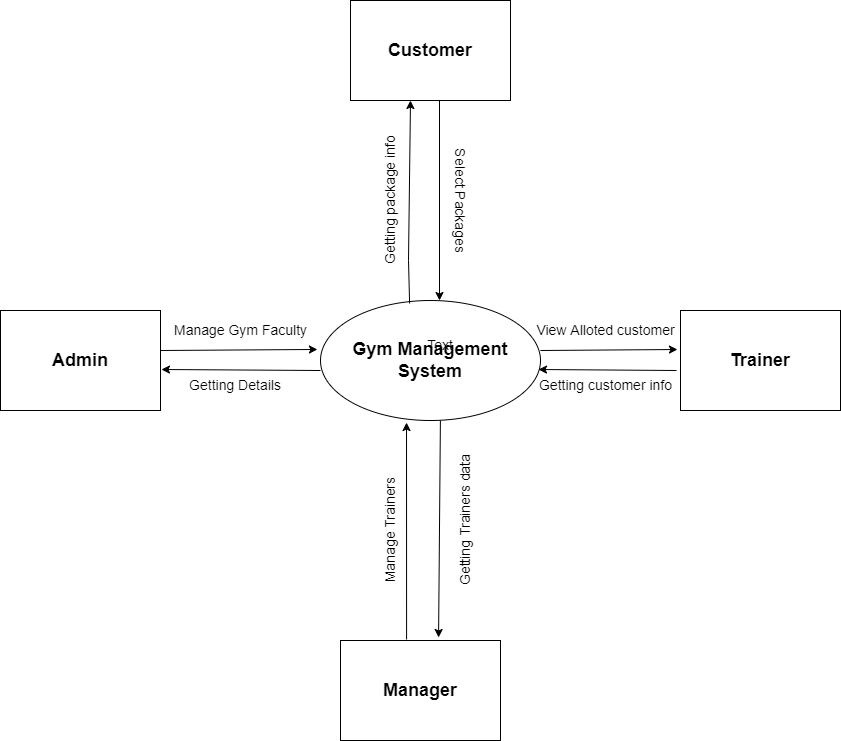


* **Admin**

Following diagram explains the page navigation for the Admin module:

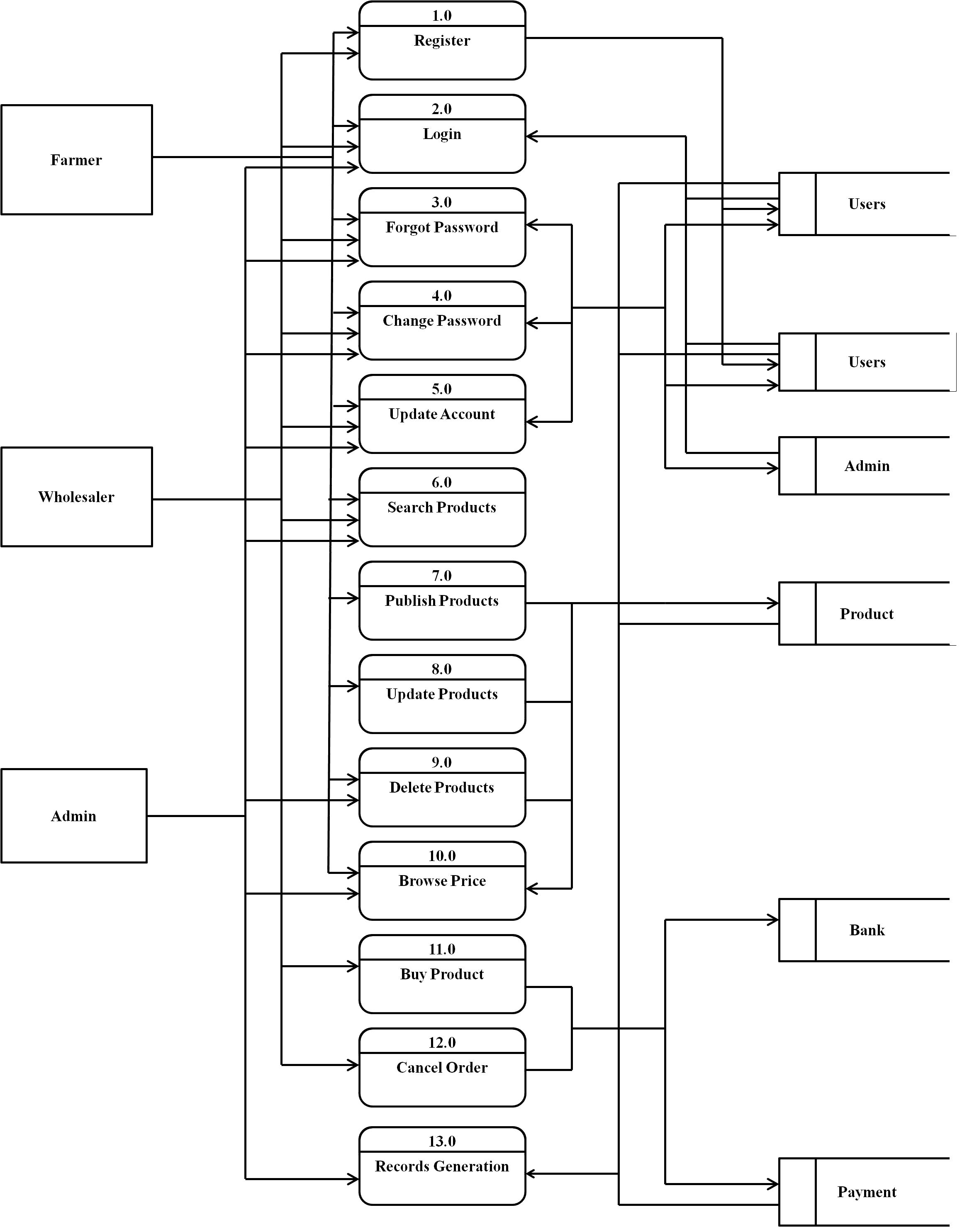


* **Data Flow Diagram:**
* **0-Level DFD:**

****

*Fig. 0-level DFD*

* In 0-Level DFD, there are four Entities:
* Manager
* Trainer
* Customer
* Admin
* **1-Level DFD:**



*Fig. 1-Level DFD*

In 1-Level DFD, Customer Entity having following processes:

* Register(Process 1.0)
* Login(Process 2.0)
* Forgot Password(Process 3.0)
* Change Password(Process 4.0)
* Update Password(Process 5.0)
* Search Package(Process 6.0)
* Apply for Membership(Process 7.0)
* View Diet and Workout plan(Process 8.0)

Trainer Entity having following processes:

* Register (Process 1.0)
* Login (Process 2.0)
* Forgot Password (Process 3.0)
* Change Password (Process 4.0)
* Update Password (Process 5.0)
* Prepare Diet Plan (Process 6.0)
* Update Diet Plan (Process 11.0)
* Browse Customer Details (Process 12.0)

Manager Entity having following processes:

* Register (Process 1.0)
* Login (Process 2.0)
* Forgot Password (Process 3.0)
* Change Password (Process 4.0)
* Update Password (Process 5.0)
* Managing Schedule (Process 6.0)
* Manage Package(Process 11.0)
* Manage Payment(Process 12.0)
* Browse Customer Details (Process 13.0)

Admin Entity having following processes:

* Login (Process 2.0)
* Forgot Password (Process 3.0)
* Change Password (Process 4.0)
* Update Password (Process 5.0)
* Access All User Information (Process 6.0)
* Hire Employee (Process 9.0)
* Pay Salary (Process 10.0)
* Release Fund (Process 14.0)
* **Deployment Diagram:**
* **Low Level Design:**
* **Database Design:**

**1] Tbl\_User**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** | **Key** | **Description** |
| UserId | Integer | No |  | Primary key | User ID |
| Fname | Varchar(45) | No |  |  | First Name |
| Lname | Varchar(45) | yes | Null |  | Last Name |
| Email | Varchar(45) | No |  | Unique | Email ID Of Customer |
| Password | Varchar(15) | No |  |  | Account Password |
| ContactNo | Varchar(15) | Yes | Null |  | Contact No. Of Customer |
| UserType | Varchar(45) | No |  |  | Customer/Trainer/Manager/Admin |
| Address | Varchar(100) | No |  |  | Permanent Address |
| Gender | Varchar(10) | No |  |  | Gender Information |
| Birthdate | Date | No |  |  | Date of Birth |

**2] Tbl\_Packages**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** | **Key** | **Description** |
| Packageid | Integer | No |  | Primary key | Trainer ID |
| Packagename | Varchar(45) | No |  |  | Package Name |
| Description | Varchar(45) | No |  |  | Package Information |
| Duration | Varchar(45) | No |  |  | Duration of Package |
| Price | Double(7,2) | No |  |  | Package price |

**3] Tbl\_CustomerPackage**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** | **Key** | **Description** |
| CustPackageid | Integer | No |  | Primary Key | Package Id of Customer |
| Packageid | Integer | No |  | Foregin Key | Reference of package |
| Userid | Integer | No |  | Foregin Key | Reference of User |
| Trainerid | Integer | No |  | Foregin Key | Reference of Trainer |
| BookingDateid | DateTime | No |  |  | Date of Booking of package |
| StartDate | DateTime | No |  |  | Start date of Package |
| EndDate | DateTime | No |  |  | End Date of Package |
| Cost | Double | No |  |  | Cost of Package |

**4] Tbl\_Payment**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** | **Key** | **Description** |
| Paymentid | Integer | No |  | Primary Key | Transaction Id |
| Status | Varchar(20) | No |  |  | Transactional approval status |
| Custpackageid | Integer | No |  | Foregin Key | Reference to custpackageid |
| Transaction mode | Varchar(20) | No |  |  | Online/Offline |
| Transaction Number | Varchar(20) | No |  |  | Transaction number |
| Date | DateTime | No |  |  | Date and time of transaction |
| ManagerId | Integer | No |  | Foregin key | Reference to ManagerId |

**5] Tbl\_CustomerDetails**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** | **Key** | **Description** |
| CustomerDetailsid | Integer | No |  |  | CustomerDetailsID |
| Userid | Varchar(45) | No |  | Foreign key | UserID |
| Date | Date | No |  |  | Date |
| Height | Integer | No |  |  | Customer Height |
| Weight | Integer | No |  |  | Customer Weight |
| Medical History | Varchar(45) | Yes | Null |  | Customer Medical History |
| Aim | Varchar(45) | No |  |  | Customer's Aim(Loose/Gain/BodyBuilding) |

* **Stored Procedure:**
* **Create Customer**:

This stored procedure is used for creating new Customer

Input parameters:

Uid, Fname, Lname, Email, Password, ContactNo, Address, Gender,

* **Details Page Navigation:**
* **Home Page:**
* Following snapshot shows the home page of the Online Gym Management system.

**Home Page Snapshot**

* This page contain following controls
* Login Button
* Registration Link
* **Login Button:**

When user fill user id and password and click login button then if the information is valid then he will be redirected to respective home page, else the error message will be shown to the user.

* **Registration Link:**

When user clicks on registration link then he will be redirect to registration page.

* **Registration Page:**
* Following snapshot shows the registration page of the Gym Management system.

**Registration Page Snapshot**

* This page contain following controls
* Submit Button
* **Submit Button:**

When user fill user information and click Submit button then if the information is valid then his account will be created and a success message will be shown to him afterwards he will be redirected to login page, else the error message will be shown to the user.

* **Customers Home Page:**
* Following snapshot shows the farmers home page of the e-farming system.

**Customers Home Page Snapshot**

* This page contain following controls
* **Search Package(Link):**

If the User clicks on the search package then he/she will be redirected to the search package page on that he can search required package information and also can go back to the home page by clicking on home page link.

* **Buy Package(Link):**

If the user click on the buy package then he will be redirected to the Buy package page on that he can buy available package and also can go back to the home page by clicking on home page link.

.

* **Update Profile(Link):**

If the user clicks on update profile then he will be redirected to update profile page, and he can change the password, update information. By clicking on save button the updated data will be store in the database. Also we can go back to the home page by clicking on home page link.

* **Browse Diet Plan(Link):**

If user Click on This he/she can browse the diet plan according to trainers instructions.

* **Trainer Home Page:**
* **Trainer Home Page Snapshot**
* Following snapshot shows the Trainers home page of the Online Gym Management system.
* This page contain following controls
* **Update Profile(Link):**

If the user clicks on update profile then he will be redirected to update profile page, and he can change the password, update information. By clicking on save button the updated data will be store in the database. Also we can go back to the home page by clicking on home page link.

* **Admin Home Page:**
* **Admin Home Page Snapshot**
* Following snapshot shows the Admin home page of the Online Gym Management system.
* This page contain following controls
* **Generate Report(Link):**

On Generate Report Page there are four Menus:

* **Today**

It shows reports for that day.

* **Weekly**

It shows all the Reports of current week.

* **Monthly**

It shows all the Reports of the current month.

* **Yearly**

It shows all the Reports of the selected year.

* **Search Product(Link):**

If the User clicks on the search Package then he will be redirected to the search package page on that he can search required product information and also can go back to the home page by clicking on home page link.

* **Browse Plan(Link):**

If the user clicks on the browse pplan then he will be redirected to the browse plan page on that he can search benifit of perticular plan and also can go back to the home page by clicking on home page link.

* **Update Profile(Link):**

If the user clicks on update profile then he will be redirected to update profile page, and he can change the password, update information. By clicking on save button the updated data will be store in the database. Also we can go back to the home page by clicking on home page link.

* **Manage Accounts(Link):**

If the user clicks on manage account then he will be redirected to manage account page, and he can manage other accounts such as he can temporarily disable any account or enable it again. Also we can go back to the home page by clicking on home page link.