Design

Document

Inter-city

Car pooling

System

**Index**

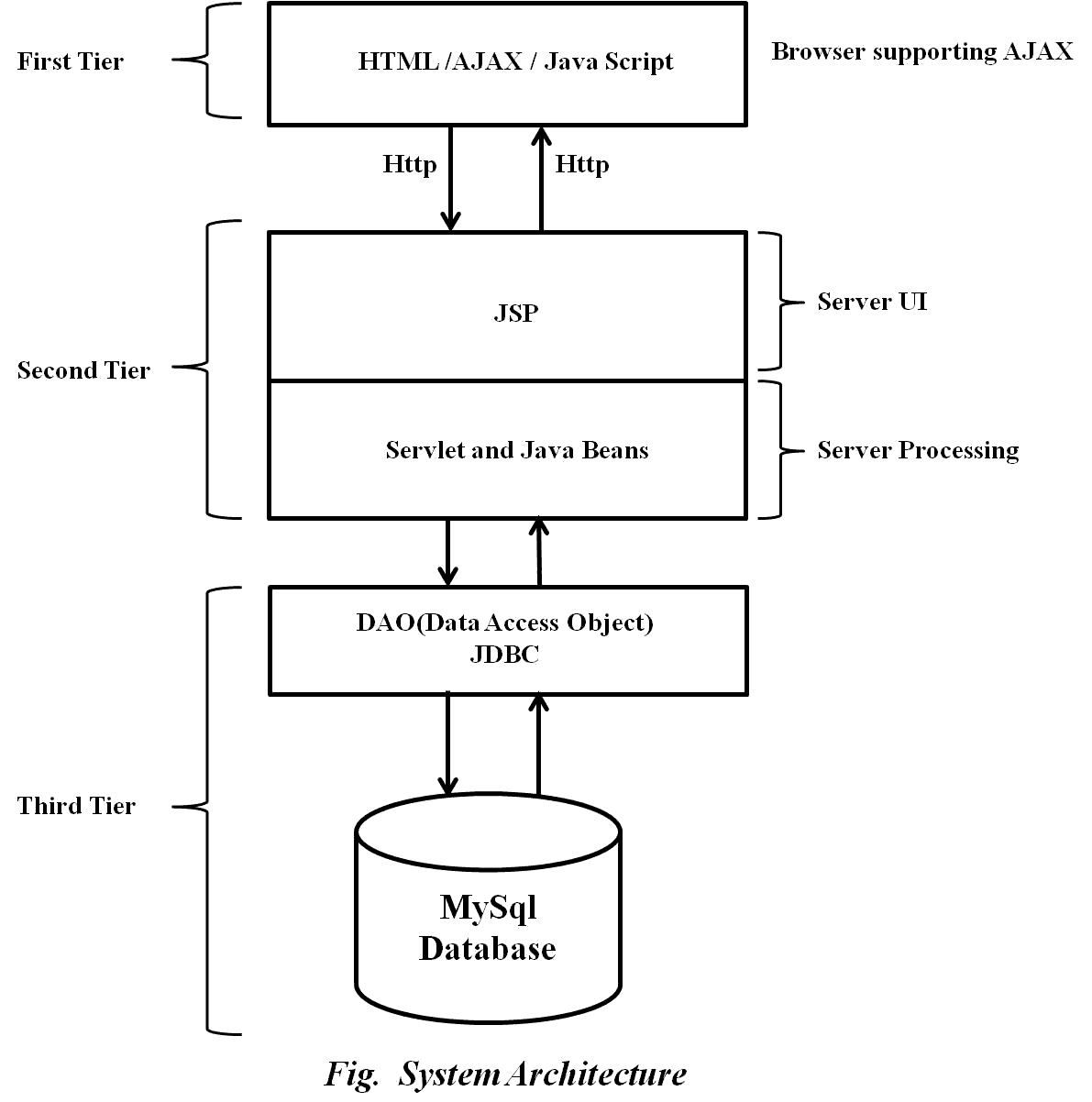
|  |  |  |
| --- | --- | --- |
| Sr No | Content | Page No |
| 1 | Introduction | 3 |
| 2 | Architectural Design | 3 |
| 3 | High Level Design | 6 |
|  | 3.1 E-R Diagram | 6 |
|  | 3.2 Page Navigation Diagram | 7 |
|  | 3.3 Data Flow Diagram | 9 |
|  | 3.4 Deployment Diagram | 12 |
| 4 | Low Level Design | 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 Inter-city Car pooling 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**

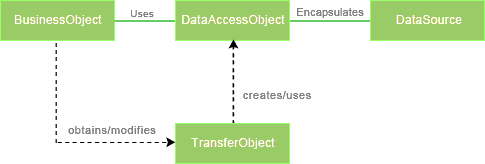
Servlet API is standard and freely available on the internet (like JSPs) servlets have the advantages like ease of development & platform independence (like Java) they can access all the J2SE and J2EE APIs can take the full advantage & capabilities of the Java programming language.

**Third Tier:**

Third tier consist of a Data Access Object (DAO) and the back end i.e. the database of Inter-city Car pooling 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.

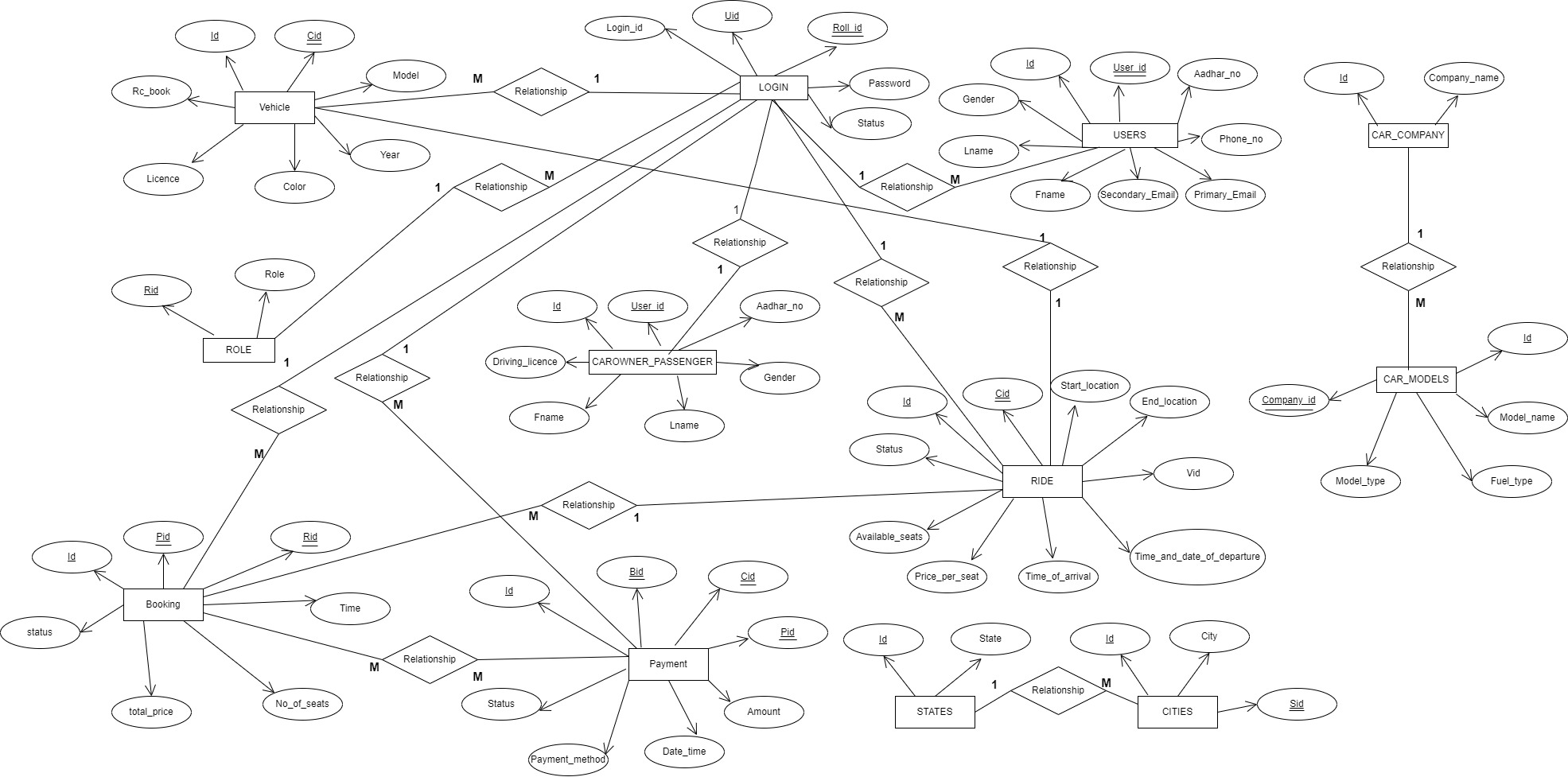


*Fig Data Access Object (DAO) Mechanism*

**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:**

* **Deployment Diagram:**
* **Low Level Design:**
* **Database Design:**

**1] Role**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Role ID |
| role | Varchar(45) | No |  | Null | Which type of user is enrolled. |

**2] Login**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Which type of user logged in |
| roll\_id | Integer | No | Foreign key | Null | Reference to roll\_id(Tbl Role) |
| login\_id | Varchar(100) | No |  | Null | Login\_id |
| password | Varchar(100) | No |  | Null | Password |
| status | Bit(1) | No |  | Null | Login Status |

**3] Users**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Car\_owner/User |
| user\_id | Integer | No | Foreign key | Null | reference to Login\_id(Tbl\_login) |
| password | Varchar(100) | No |  | Null | Password of user |
| aadhar\_no | Varchar(45) | No |  | Null | Aadhar number of user |
| fname | Varchar(100) | No |  | Null | First name of User |
| lname | Varchar(100) | No |  | Null | Last name of User |
| gender | Varchar(45) | No |  | Null | Gender of User |
| dob | Date | No |  | Null | Date of Birth of User |
| licence | Varchar(45) |  |  |  |  |
| phone-no | Varchar(45) | No |  | Null | Phone no of User |
| primary email | Varchar(100) | Yes |  | Null | email id of User for login |
| secondary\_email | Varchar(100) | Yes |  | Null | Secondary email id of User |

**4] co\_passenger**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Co-passenger ID |
| passenger\_id | Integer | No | Foreign key | Null | Reference to User\_id (Tbl\_User) |
| aadhar\_no | Varchar(45) | Yes |  | Null | Aadhar no of co\_passenger |
| phone\_no | Varchar(45) | Yes |  | Null | Phone no of co\_passenger |
| email | Varchar(45) | Yes |  | Null | Email of co\_passenger |
| fname | Varchar(45) | No |  | Null | First name of co\_passenger |
| lname | Varchar(45) | No |  | Null | Last name of co\_passenger |
| Gender | Varchar(45) | No |  | Null | Gender of co\_passenger |

**5] Car\_company**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Admin ID |
| company\_name | Varchar(100) | No |  | Null | Company name of car |

**6] Car\_Models**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Admin ID |
| Company\_id | Integer | No | Foreign\_key | Null | Reference to Car\_company\_id(Tbl\_**Car\_company**) |
| model\_name | Varchar(100) | No |  | Null | Model name of car |
| fuel\_type | Varchar(100) | No |  | Null | Fuel type of car |
| model\_type | Varchar(100) | No |  | Null | Model type of car |

**7] Vehicles**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Vehicle id |
| carowner\_id | Integer | No | Foreign key | Null | Reference to Users\_id(Tbl\_**users**) |
| model\_id | Integer | No | Foreign key | Null | Reference to Car\_Model\_id(Tbl\_c**ar\_model\_**) |
| year | Integer | No |  | Null | Year of buying car |
| color | Varchar(45) | No |  | Null | Color of car |
| rc-book | Varchar(200) | No |  | Null | Registration card of car owner |

**8] Rides**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Ride Id |
| carowner\_id | Integer | No | Foreign key | Null | Reference to Carowner\_id(Users\_Tbl) |
| start\_Location | Integer | No | Foreign key | Null | Reference to Cid (Tbl\_City) |
| end\_Location | Integer | No | Foreign key | Null | Reference to Cid (Tbl\_City) |
| vehicle\_id | Integer | No | Foreign key | Null | Reference to Vid (Tbl\_Vehicle) |
| time\_and \_date\_of\_departure | DateTime | No |  | Null | Time and date of departure |
| time\_of\_arival | datetime | No |  | Null | Arrival time |
| price\_per\_seat | Varchar(100) | No |  | Null | Price per seat amount |
| Available\_seats | Varchar(100) | No |  | Null | Total available seats in car |
| status | Varchar(100) | Yes |  | Null | Ride is confirmed or not |

**9] Booking**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Booking ID |
| passenger\_id | Integer | No | Foreign key | Null | Reference to passenger\_id(Tbl\_User) |
| ride\_id | Integer | No | Foreign key | Null | Reference to Ride\_id(Tbl\_Ride) |
| time | DateTime | No |  | Null | Booking time |
| no\_of\_seats | Integer | No |  | Null | How many seats are booked |
| total\_price | Integer | No |  | Null | Total price acc to passenger |
| status | Varchar(100) | No |  | Null | Status of booking confirmation or not |

**10] Payment**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | Payment ID |
| booking\_id | Integer | No | Foreign key | Null | Reference to Booking\_id(Tbl\_Booking) |
| passenger\_id | Integer | No | Foreign key | Null | Reference to passenger\_id(Tbl\_Co\_passenger) |
| amount | Integer | No |  | Null | Total amount |
| Date\_Time | datetime | No |  | Null | Date and time of payment |
| payment\_method | Varchar(45) | No |  | Null | Permanent method |
| status | Varchar(100) | No |  | Null | Paymnt is succesfull or not |

**11] State**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | State id |
| state | Varchar(45) | No |  | Null | State name |

**12] City**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | City id |
| city | Varchar(45) | No |  | Null | City name |
| state\_id | Integer | No |  | Null | Reference to State\_id(Tbl\_State) |

**13] Passenger\_review**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Integer | No | Primary key | Null | City id |
| passenger\_id | Integer | No | Foreign key | Null | Reference to User\_id(Tbl\_User) |
| ride\_id | Integer | No | Foreign key | Null | Reference to Ride\_id(Tbl\_Ride) |
| rating | Decimal(3,2) | No |  | Null | Rating given by user to car\_owner. |
| comments | Varchar(45) | No |  | Null | Any comments. |

* **Stored Procedure:**
* **CreateUser**:

This stored procedure is used for creating new user

Input parameters:

Uid, Fname, Lname, Email, Password, ContactNo, Address, Gender, AccNo, Image, Type

* **SendMessage:**

This stored procedure is used for inserting message details

Input parameters:

ReceiverId, SenderId, Message, Subject

* **Details Page Navigation:**
* **Home Page:**
* Following snapshot shows the home page of the Inter-city Car pooling 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 e-farming 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.

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

**Farmers Home Page Snapshot**

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

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

* **Publish Product(Link):**

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

* **Update Product(Link):**

If the user clicks on the update product then he will be redirected to the update product page on that he can update product information as required and also can go back to the home page by clicking on home page link.

* **Delete Product(Link):**

If the user clicks on the delete product then he will be redirected to the delete product page on that he can delete product information and also can go back to the home page by clicking on home page link.

* **Browse Price(Link):**

If the user clicks on the browse price then he will be redirected to the browse price page on that he can search current market price of a particular product 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.

* **Messages(Link):**

If the user clicks on Message then he will be redirected to the messages page there will be three option links,

* **Compose Message**

On clicking Compose Message link Farmer can create the message and redirected to the Admin or wholesaler.

* **Inbox**

It will display all the Messages he has received.

* **Outbox**

It will display all the sent messages.

* **Wholesaler Home Page:**
* **Wholesaler Home Page Snapshot**
* Following snapshot shows the farmers home page of the e-farming system.
* This page contain following controls
* **Search Product(Link):**

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

* **Buy Product(Link):**

If the user clicks on the buy product then he will be redirected to the buy product page on that he can search required product and can buy it. Also we can go back to the home page by clicking on home page link.

* **Payback(Link):**

In case If the wholesaler cannot satisfied with the Product. He can click on the payback then he will be redirected to the payback page on which he can select his unsatisfied transaction and click on payback button and his request will be redirected to the admin. Also we can go back to the home page by clicking on home page link.

* **Cancel Order(Link):**

If the user clicks on the cancel order then he will be redirected to the cancel order page on that he can cancel the order. Also we can go back to the home page by clicking on home page link.

* **Browse Price(Link):**

If the user clicks on the browse price then he will be redirected to the browse price page on that he can search current market price of a particular product 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.

* **Messages(Link):**

If the user clicks on Message then he will be redirected to the messages page there will be three option links,

* **Compose Message**

On clicking Compose Message link Farmer can create the message and redirected to the Admin or wholesaler.

* **Inbox**

It will display all the Messages he has received.

* **Outbox**

It will display all the sent messages.

* **Wholesaler Home Page:**
* **Admin Home Page Snapshot**
* Following snapshot shows the Admin home page of the e-farming 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 product then he will be redirected to the search product 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 Price(Link):**

If the user clicks on the browse price then he will be redirected to the browse price page on that he can search current market price of a particular product 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.

* **Messages(Link):**

If the user clicks on Message then he will be redirected to the messages page there will be three option links,

* **Compose Message**

On clicking Compose Message link Farmer can create the message and redirected to the Admin or wholesaler.

* **Inbox**

It will display all the Messages he has received.

* **Outbox**

It will display all the sent messages.