**Project Profile**

**Existing System**

These kind of system are usually less in existence, and if they exists then it is not easily known. The admin has already a school but now needs to bring into existence a Daycare too.

**Proposed System**

This system will be surely useful later as per the scenario of today’s generation. So it is a better and a wiser step to implement this from now. Thus there is a need for this system. The parent’s can leave their kids here during their work hours and kids are given here home like environment and treated as if their home only. The kids here will be provided games, toys, food, different materials, etc.

**ADMIN:**

* Admin will manage inquiry and registration from parents.
* Admin will add activities to be done and manage the materials.
* Different packages are kept for different demands of parents.
* Admin adds photos of the kids so that parents can view it.
* Payment will be made to admin.
* At last feedback will be viewed by admin.

**Caretaker:**

* The activities whichever are done by children’s in their daily routines is to be seen by care taker.
* Whether the child is daily present or not is to be taken care off by caretaker.
* A schedule is to be arranged for meeting parents by caretaker only to discuss about the kids progress.
* At the end of every week, month, or year a report of each children is generated so that their performance can be known by their parents.

**Parents:**

* Parents can view day care details and can choose if it is preferable to them.
* After registration they can purchase any package for how much time they need to keep their child in daycare.
* The parents can meet the respective care taker and can know about their children’s performance.
* The report which is generated by care taker can be viewed by parent’s.
* The parents can give feedback and rate the day care.

**Technology Used**

**Front End** :-

PHP (MVC)

**Back End** :-

Database:My SQL

Web Server:Apache

PHP:

* PHP (acronym for PHP: “Hypertext Preprocessor") is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.
* PHP is an **interpreted** language, in that the PHP interpreter program reads the PHP source code, translates the code and executes it at the same time.
* PHP was developed to specifically address needs of the web to provide dynamic content on websites.
* PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.
* PHP is a widely used, free, and efficient alternative to competitors such as Microsoft’s ASP.
* PHP can generate dynamic page content. It can create, open, read, write, delete, and close files on the server.
* PHP can send and receive cookies.
* PHP can add, delete, modify data in your database. It can restrict users to access some pages on your website.

MySQL:

MySQL is world’s second most widely used open-source relational database management system. It is named after co-founder Michael Widenious’s daughter.

MySQL, pronounced either "My S-Q-L" or "My Sequel," is an open source relational database management system. It is based on the structure query language ([SQL](http://techterms.com/definition/sql)), which is used for adding, removing, and modifying information in the database.

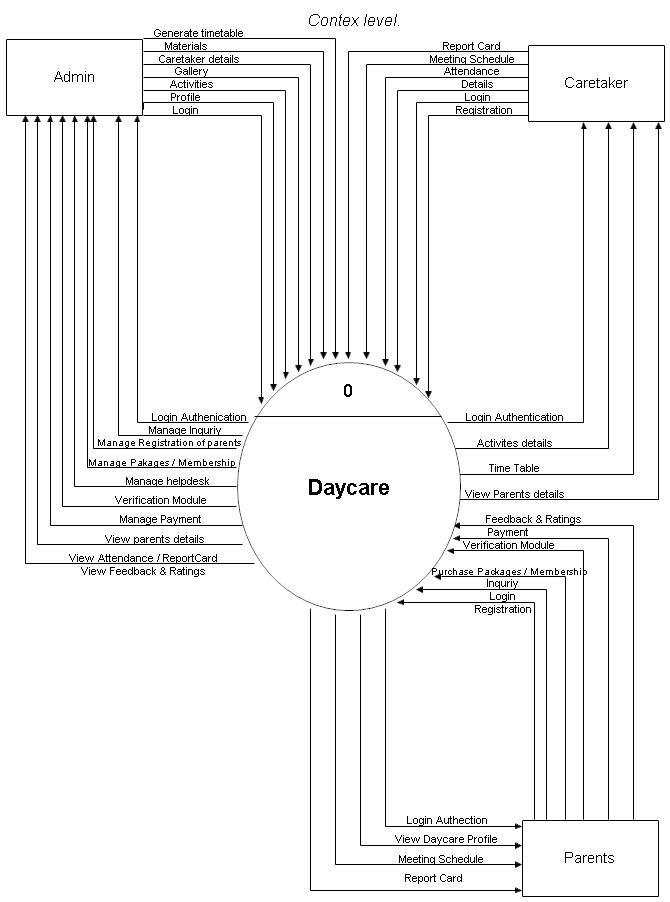
MySQL can be used for a variety of applications, but is most commonly found on Web servers.

MySQL commands can be incorporated into the PHP code, allowing part or all of a Web page to be generated from database information. Because both MySQL and PHP are both open source (meaning they are free to download and use), the PHP/MySQL combination has become a popular choice for database-driven websites.

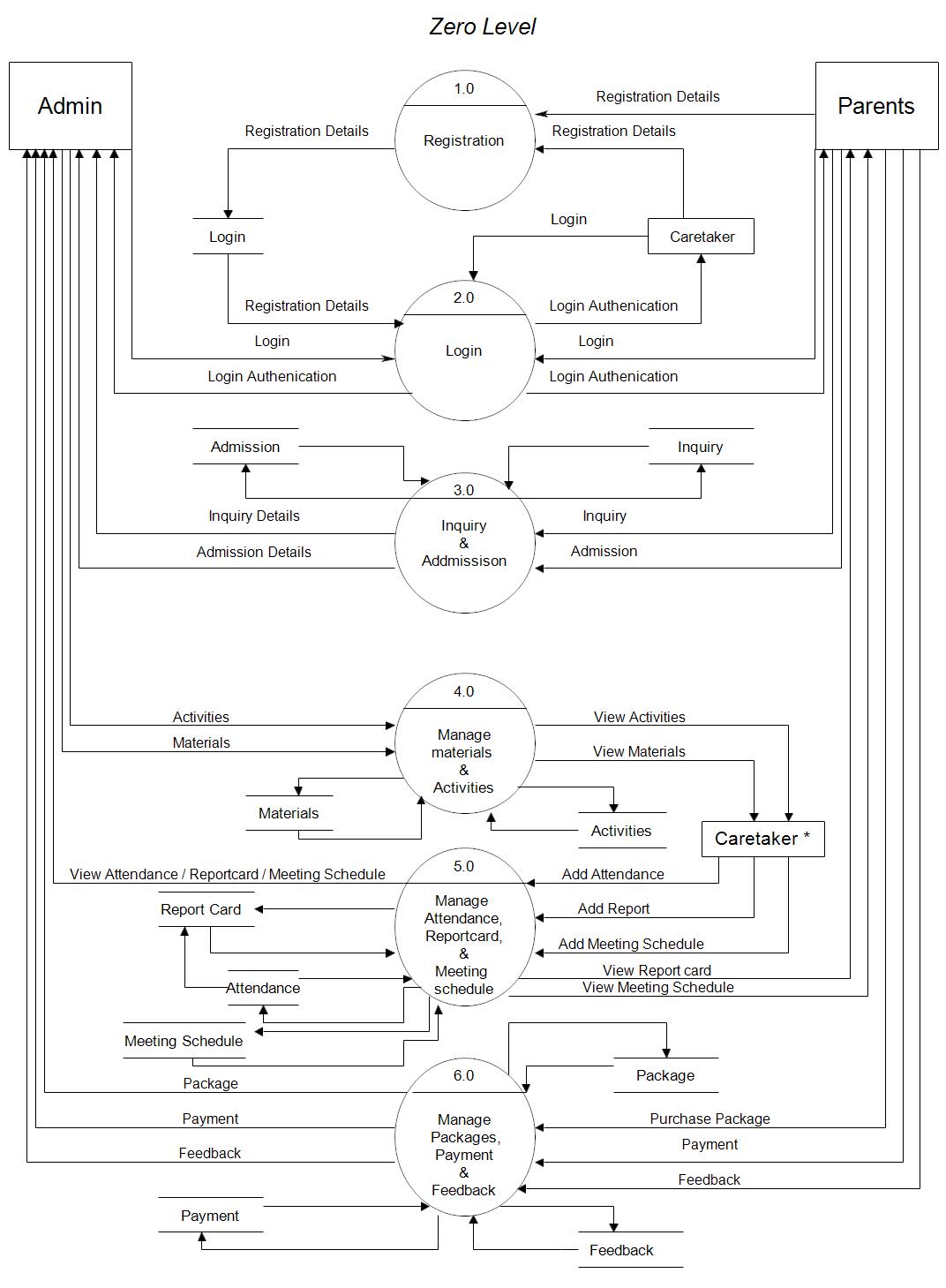
Many of the world’s fastest and largest growing organizations use MySQL to save time and money powering their high-volume websites, critical business system, and packaged software including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, and Wikipedia.

DATA FLOW DIAGRAMS

CONTEXT LEVEL DIAGRAM

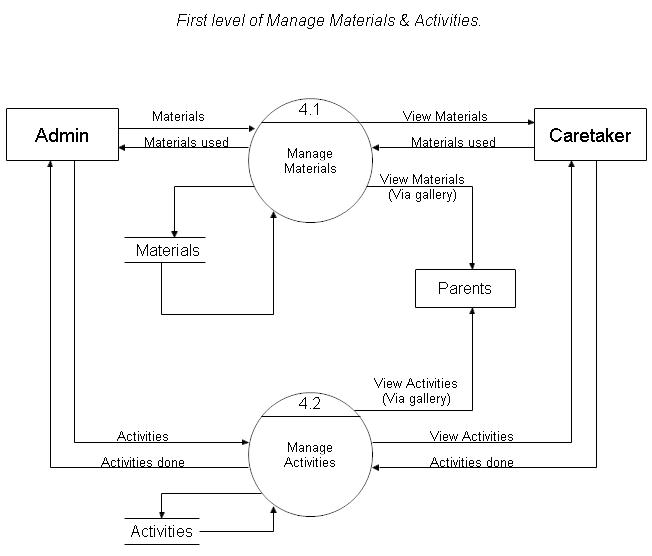
****

ZERO LEVEL DIAGRAM

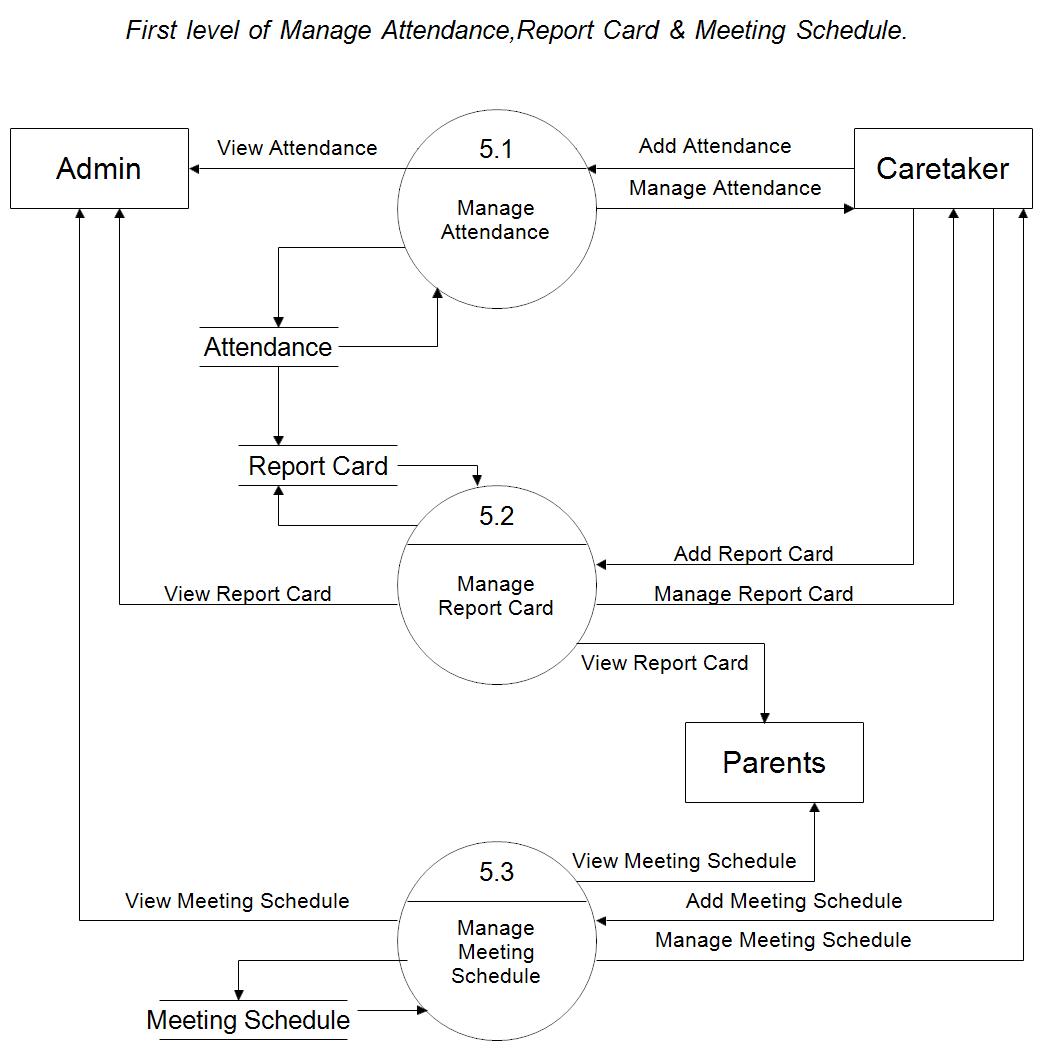
****

FIRST LEVEL DIAGRAMS

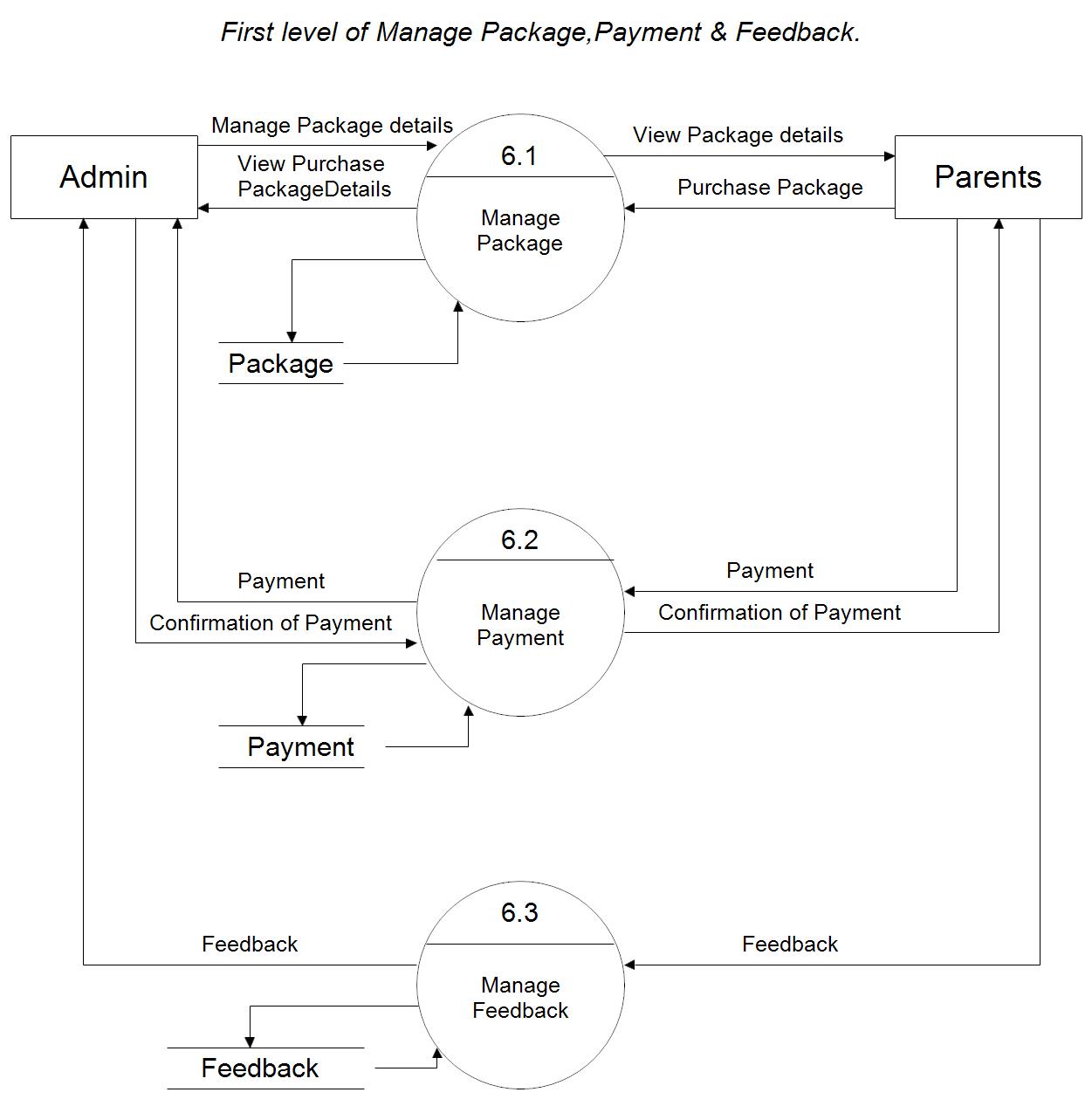
FIRST LEVEL FOR MANAGE MATERIALS & ACTIVITIES

****

FIRST LEVEL FOR MANAGE ATTENDANCE, REPORTCARD & MEETING SCHEDULE



FIRST LEVEL FOR MANAGE PACKAGE PAYMENT & FEEDBACK

****

DATA DICTIONARY

|  |  |
| --- | --- |
| **No.** | **Table Names** |
| **1** | Admin |
| **2** | Area |
| **3** | Inquiry |
| **4** | Registration |
| **5** | Package |
| **6** | Admission |
| **7** | Care taker |
| **8** | Materials |
| **9** | Activity |
| **10** | Attendance |
| **11** | Report Card |
| **12** | Meeting |
| **13** | Fees |
| **14** | Feedback |

Table 1: **ADMIN**

This gives information about Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Admin\_id | Int | 2 | Unique id of admin | PK(Auto increment) |
| Admin\_name | Varchar | 12 | Name of admin | - |
| Email | Varchar | 25 | Email of admin | - |
| Password | Varchar | 12 | Password of admin | - |

Table 2: **AREA**

This gives information that from which area all are located

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Area\_id | Int | 3 | Unique id of each area | PK(Auto increment) |
| Area\_name | Varchar | 10 | Name of each area | - |

Table 3: **INQUIRY**

This gives information about inquiry done

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Inq\_id | Int | 3 | Unique id of each inquiry | PK(Auto increment) |
| Area\_id | Int | 3 | Unique id of each area | FK(Area Table) |
| Name | Varchar | 12 | Name of the person inquiring | - |
| Address | Varchar | 50 | Address of the person inquiring | - |
| Contact no | Int | 10 | Contact of the person inquiring | - |
| Inq\_details | Varchar | 100 | From where inquiry is done | - |
| Date | Date | - | Date of inquiry | - |
| Email\_id | Varchar | 30 | Email of inquring person | - |
| Status | Int | 1 | Status of inquiry | - |

Table 4: **REGISTRATION**

This gives information about the registration.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Reg\_id | Int | 3 | Unique id of each registration | PK(Auto increment) |
| Area\_id | Int | 3 | Unique id of each area | FK(Area Table) |
| Last\_name | Varchar | 10 | Last name of person | - |
| First\_name | Varchar | 10 | First name of person | - |
| Address | Varchar | 50 | Address of person | - |
| Contact no. | Int | 10 | Contact no of person | - |
| Relationship with chid | Varchar | 10 | The person’s relation with child | - |
| Photo | Varchar | 500 | Photo pf person being registered | - |

Table 5: **PACKAGE**

This gives information about different packages.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Pack\_id | Int | 3 | Unique id of each package | PK(Auto increment) |
| Pack\_name | Varchar | 10 | Name of package | - |
| Duration | Int | 2 | Duration of package | - |
| Fees | Int | 5 | Fees of package | - |
| Installment | Enum | - | Installment of package(True/False) | - |
| Food Facility | Enum | - | Whether the facility is available or not in package | - |
| Age factor | Enum | - | The age of child included in package or not | - |
| Time factor | Enum | - | The package is valid according to time | - |
| Academic Education | Enum | - | Whether the child needs education facility or not | - |

Table 6: **ADMISSION**

This gives information about admission

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Ad\_id | Int | 3 | Unique id of each admission | PK(Auto increment) |
| Pack\_id | Int | 3 | Unique id of each package | FK(Package Table) |
| Reg\_id | Int | 3 | Unique id of each registration | FK(Registration Table) |
| Name of child | Varchar | 12 | Name of child for admission | - |
| Name of parents/guardian | Varchar | 12 | Name of child’s parents/guardian | - |
| D.O.B | Date | - | Date of birth of child | - |
| Address of child | Varchar | 50 | Address of child | - |
| Contact no | Int | 10 | Contact no of parents | - |
| Emergency contact no. | Int | 10 | Number of person to be contacted in emergency | - |
| Mode of transportation | Varchar | 10 | Mode of transportation how child will come | - |
| Email id of parents | Varchar | 25 | Email\_id of parents | - |
| Medical problem with child | Enum | - | Yes/No | - |
| Details of medical problem | Varchar | 50 | Details if child is having any medical problem | - |
| Disability of child | Enum | - | Yes/No | - |
| Details of disability of child | Varchar | 50 | Details if child is having any disability | - |

Table 7: **CARE\_TAKER**

This gives information about care taker.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Ct\_id | Int | 3 | Unique id of each caretaker | PK(Auto increment) |
| area\_id | Int | 3 | Unique id of each area | FK(area Table) |
| Ct\_name | Varchar | 20 | Name of caretaker | - |
| Email | Varchar | 25 | Email id of care taker | - |
| Password | Varchar | 12 | Password of care taker | - |
| Gender | Enum | - | Male/Female | - |
| Address | Varchar | 50 | Address of care taker | - |
| Contact no. | Int | 10 | Contact no. of care taker | - |
| D.O.J | Date | - | Date of joining of care taker | - |

Table 8: **MATERIALS**

This gives information about materials.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Mat\_id | Int | 3 | Unique id of each material | PK(Auto increment) |
| Mat\_name | Varchar | 15 | Name of materials | - |
| Quantity | Int | 2 | Quantity of materials | - |
| Current\_in\_use | Int | 2 | Materials being used | - |
| Status | Int | 1 | Status of inquiry | - |

Table 9: **ACTIVITY**

This gives information about materials.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Act\_id | Int | 3 | Unique id of each activity | PK(Auto increment) |
| Act\_name | Varchar | 20 | Name of activity | - |

Table 10:**ATTENDANCE**

This gives information about attendance.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Att\_id | Int | 3 | Unique id of each attendance | PK(Auto increment) |
| Ct\_id | Int | 3 | Unique id of each caretaker | FK(Caretaker Table) |
| Ad\_id | Int | 3 | Unique id of each admission | FK(Admission Table) |
| Date | Date | - | Date of attendance | - |
| Status | Enum | - | Present/Absent | - |

Table 11: **REPORT\_CARD**

This gives information about report card.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Rep\_id | Int | 3 | Unique id of each report card | PK(Auto increment) |
| Rep\_details | Varchar | 200 | Details of report | - |
| Total\_Attendance | Int | 3 | Attendance of child throughout the month | - |
| Ad\_id | Int | 3 | Unique id of each admission | FK(Admission Table) |
| Ct\_id | Int | 3 | Unique id of each caretaker | FK(Caretaker Table) |

Table 12: **MEETING**

This gives information about meeting schedule.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Meet\_id | Int | 3 | Unique id of each meeting | PK(Auto increment) |
| Ct\_id | Int | 3 | Unique id of each caretaker | FK(Caretaker Table) |
| Meet\_agenda | Varchar | 30 | Topic of meeting | - |
| Date/Time | Date | - | Date and time of meeting | - |

Table 13: **FEES**

This gives information about fees structure.

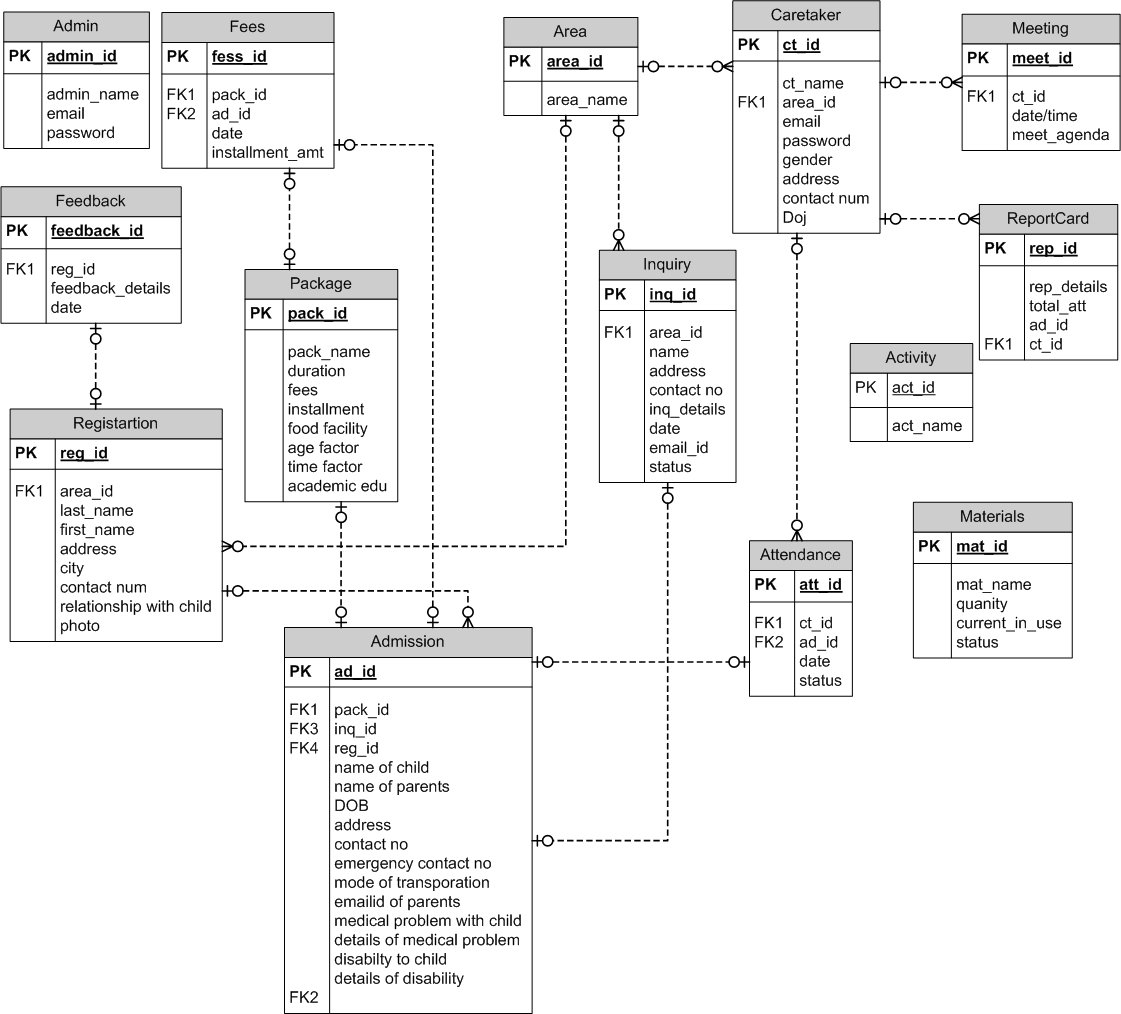
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Fees\_id | Int | 3 | Unique id of each fees | PK(Auto increment) |
| Ad\_id | Int | 3 | Unique id of each admission | FK(Admission Table) |
| Date | Date | - | Date of fees | - |
| Installment\_amt | Int | 5 | Amount of installment | - |

Table 14: **FEEDBACK**

This gives information about feedback.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Size | Description | Constraint |
| Feedback\_id | Int | 3 | Unique id of each feedback | PK(Auto increment) |
| Reg\_id | Int | 3 | Unique id of each registration | FK(Registration Table) |
| Feedback details | Varchar | 200 | Feedback details to be stored | - |
| Date | Date | - | Date of feedback given | - |

ERD DIAGRAM



**Bibliography**

Reference Book:-

1. MagnifyingObject-OrientedAnalysisAndDesign.

Authors:-Arpita Gopal And Netra Patil

Publisher:- PHP Learning

Internet:-

<www.W3schools.com>

Conclusion

At the end of our project we conclude that it was a really good experience to make the website for “Daddy Daycare” under the guidance of our Internal Guide “TriptiDodiya”. We are thankful to her for providing the guidance and helping us in our project.