ANSAR COMPUTER CENTER

Perumpilavu, Karikkad (p.o),Thrissur, Kerala- 680519

PROJECT REPORT

ON

"KIDDIE CLOUD"

Submitted By:

MUFITHA MUSTHAFA

(Reg.No:944857)

Under the guidance of

Mrs. Nusaiba

Submitted in partial fulfillment of the requirements for the award of

A'LEVEL

National Institute of Electronics and Information Technology
Department of Electronics and Information Technology
Government of India
Electronics Niketan, 6, CGO Complex
New Delhi 110 003

DECLARATION

I here by declare that the project entitled "***KIDDIE CLOUD***" submitted in partial fulfillment of requirements for A' Level of NIELIT, is my original work under the supervision and guidance of Mrs. Nusaiba. And it is not submitted for the award of any other degree, diploma, fellowship or any other similar title or prizes. The empirical findings in the report are based on information collected by me and not copied from elsewhere.

I understand that detection of any such copying is liable to be punished in any way the School deems fit.

Place: Perumpilavu
Date:

MUFITHA MUSTHAFA
Reg.No: 944857

ACKNOWLEDGEMENT

"It is not possible to prepare a project report without the assistance & encouragement of other people. This one is certainly no exception."

At the very outset of this report, I would like to extend my sincere & heartfelt obligation towards all the personages who have helped me in this endeavor. Without their active guidance, help, cooperation & encouragement, I would not have made headway in the project.

I would like to state my special thanks to center manager, Mrs. Latha for providing the facilities and motivating me for doing this, I may note down special thanks to Mrs. Nusaiba, for stimulating her suggestions and encouragement, and to all the staff members of my college. I would like to appreciate the guidance given by other supervisors as well as the panels especially in our project presentation that has improved our presentation skills by their comment and tips.

MUFITHA MUSTHAFA

ABSTRACT

This is a web portal which mainly aims at parenting and childcare. This system concentrates on children in the age group of 1 to10.Their vaccination, food, diet and nutrition details where all will be able from this system. Parents can even book for pediatricians. This will be very helpful system for the parents to take care their children with proper guidance. They can send their doubt to admin and will get replied.

Purpose

The purpose of developing Kiddie Colud website is to computerized the tradition way of maintaining their activities. Another purpose for developing this website is to book for the doctors by theparents for their children.

Scope Of The Project

The scope of the project is the system on which the project is installed, i.e. the project is developed as a website, and it will work for a particular Organization. But later on the project can be modified to operate it online.

CONTENTS

INTRODUCTION

1.1 INTRODUCTION

This is a web portal which mainly aims at parenting and child care. This system concentrates on children in the age group of 1 to 10. Their vaccination, food, diet and nutrition details where all will be available from this system. Parents can even book for pediatricians. This will be very helpful system for the parents to take care their children with proper guidence. They can sent their doubts to admin and will get replied. Child care may be defined as care for young children, provided by adults who are not their parents. Informal child care by relatives, nannies, or home care providers typically takes place in a home setting, while formal care by trained and untrained caregivers takes place in school or care center setting.

Child care is now an ordinary part of life for children in most western countries. More than half of infants are placed in some form of child care for at least ten hours during their first years of life, and more than three-quarters of families with young children depend on child care as a support for maternal employment.

Child Crae software is a standalone application which has features like

- Adding/Deleting/Updating doctors
- Adding/Deleting/Updating speakers
- Adding/Deleting/Updating Child products
- Booking for doctors
- Booking for Speakers
- Adding/Deleting/Updating Vaccination details
- Registration for Parents
- Adding/Deleting/Updating Child right and laws
- Adding/Deleting/Updating Tips

MODULES:

Child Care System has mainly four modules

- Administrator Module

Admin manages the whole system. They can add all details of child rights and laws. Add food and vaccination details for each age group. Add pediatricians and provide login. Add schools and their related details. Add parenting tips. Add details of baby care products and shops. Rural area helping facilities. View all the complaints and take essential remedies about child abuse from users. Add motivational speakers and social activists to the system 50 admin can arrange sessions or other facilities as per the request of users especially from users. View booking for activists/motivational speakers. View other queries from users. Add baby shops and baby care products.

- Doctor Module

Doctor can login to the system. View their bookings and adjust time slots. Approve bookings. Share their ideas and tips about parenting. How to take care children of all age group and tips for the physical and mental well being of the children. Views queries from users and replays.

- Parent Module

Register and login. View all the details added by admin. View and book pediatricians. Search schools by locations. Register complaints. View baby care products and shops. View parenting tips, child care details, add talents of their children. Ask queries to doctors and view speakers &activists.

- Public Module

View activists and motivational speakers, get notified on approval, view tips about parenting child care. Register complaints to admin about child abuses etc

.

SYSTEM ANALYSIS AND DESIGN

2.1 STUDY PHASE

In Gimnasiums the works may be done manually. If it is computerized there will be no chance for errors. Moreover storing and retrieving of information is easy. So work can be done speedily and in time.

2.2 EXISTING SYSTEM

In earlier the child care was not developed. We deeply believe that, as Canadians and global citizens, we can and must continue to do more to help children who are living in poverty. Preparing facilities for child care is a comprehensive design to help evaluate the redness and to prepare existing facilities to house. Quality children care programs for children age from 1 to 10. In some cases properties may also display effects of deferred or neglected maintenance.With an existing facility in order to propose it for occupancy by a child care program**.**

2.1.1 DRAWBACK

- To increase cost.
- Burden of paper work.
- Wastage of time

2.3 PROPOSED SYSTEM

This is a web portal which mainly aims at parenting and child care. This system concentrates on children in the age group of 1 to 10. Their vaccination, food, diet and nutrition details will be very helpful system for the parents to take care their children with proper guidance get replied. The current system still using paper to record all the information thus it might least to occurrence of data loss by using the proposed system staff can store the

information systemically and they can perform backup services provided in the system. There for, if the data loss occurs, they can recover back the data in a short time.

2.4 FEASIBILITY STUDY

Feasibility is carried out to select the best system that meets performance requirement. In case the system proposal is accepted to the management, the next phase is to examine the feasibility of the system. The feasibility study is basically the test of the proposed system in the light of its working, meeting user's requirement, effective use of resources and the cost effectiveness. These are categorized as technical, operational , economical , schedule and social feasibility .

In the process of feasibility study , the cost and benefits are estimated with greater accuracy to find the return on Investment (ROI) . This also defines the resources needed to complete A Feasibility study is carried out to select the best system that meets performance the detailed investigation. The result is a feasibility report submitted to the management. This may be accepted or accepted with modifications or reject.

The Main aim of the feasibility study is to determine whether it would be financially and technically feasible to develop the product. The feasibility study activity involves the analysis of the problem and collection of all relevant information related to the product such as the different data item which would be input to the system , the processing requirement to be carried out on these data, the output data required to be produced by the system as well as various constraints on the behavior of the system.

Steps in feasibility study :

(a) **Technical feasibility**: The first technique in the feasibility analysis is to assess the technical feasibility of the project and the extend to which the system can be successfully designed , developed and installed by IT group . This is concerned with specifying equipment and software that will successfully satisfy the user requirement. This involves financial consideration to accommodate technical enhancement. If the budget is a serious constraint then the project is judged as not feasible.

(b) **Economic feasibility**: The second element of a feasibility analysis is to perform an economic feasibility analysis (also called a cost-benefit analysis) that identifies the financial risk associated with the project. This attempts to answer the question "should we build the system?" Economic feasibility is determined by identifying costs and benefits associated with the system , assigning values to them, and then calculating the cash flow and return on investment for the project. The more expensive the project the more rigorous and detailed the analysis should be. There must be scopes for profit after the successful completion of the project.

(c) **Operational feasibility** : Operational feasibility is concerned with human , organizational and political aspects. Operational feasibility covers two aspects . one is the technical performance aspect and the other is acceptance within the organization. Technical includes issues such as determining whether the system can provide the right information for the organization's personnel, and whether the system can be organized so that it always delivers this information at the right place and at right time.

SYSTEM CONFIGURATIONS

3.1 HARDWARE SPECIFICATION:

PROCESSOR	:	Pentium(R) Dual-Core CPU
CLOCK SPEED	:	500MHZ
SYSTEM BUS	:	32 BIT
RAM	:	4 GB
HDD	:	20 GB
MONITOR	:	SVGA COLOR
KEY BOARD	:	STANDARD 101/102-KEY
MOUSE	:	HID-COMPLAINT MOUSE
FDD	:	1.44 MB’

3.2 SOFTWARE REQUIREMENTS:

OPERATING SYSTEM	:	MICROSOFT WINDOWS 10 PROFESSIONAL
ENVIRONMENT	:	Php Codeigniter
FRONT END	:	Php , Html
BACK END	:	MySql

3.3 FEATURES OF SOFTWARE USED

About Windows 10 Professional

Windows 10 is a series of personal computer operating systems produced by Microsoft as part of its Windows NT family of operating systems. It is the successor to Windows 8.1, and was released to manufacturing on July 15, 2015, and broadly released for retail sale on July 29, 2015.[9] Windows 10 receives new builds on an ongoing basis, which are available at no additional cost to users, in addition to additional test builds of Windows 10 which are available to Windows Insiders. Devices in enterprise environments can receive these updates at a slower pace, or use long-term support milestones that only receive critical updates, such as security patches, over their ten-year lifespan of extended support.[10][11]

One of Windows 10's most notable features is support for universal apps, an expansion of the Metro-style apps first introduced in Windows 8. Universal apps can be designed to run across multiple Microsoft product families with nearly identical code—including PCs, tablets, smartphones, embedded systems, Xbox One, Surface Hub and Mixed Reality. The Windows user interface was revised to handle transitions between a mouse-oriented interface and a touchscreen-optimized interface based on available input devices—particularly on 2-in-1 PCs, both interfaces include an updated Start menu which incorporates elements of Windows 7's traditional Start menu with the tiles of Windows 8. Windows 10 also introduced the Microsoft Edge web browser, a virtual desktop system, a window and desktop management feature called Task View, support for fingerprint and face recognition login, new security features for enterprise environments, and DirectX 12.

Windows 10 Troubleshooting

Like any other software Windows 10 comes with many bug, undetected during the testing phase. In case the users face any problem while using it, online computer support should be sought. If you're using Windows 10 or Windows 8, you'll notice that you don't have a service pack installed. This is because, with these versions of Windows, Microsoft releases updates on

a continual basis in smaller chunks instead of infrequent, large packs as was the case with earlier Windows versions.

Introduction to php

PHP is a general purpose server-side scripting language originally designed for Web development to produce dynamic Web pages. It is one of the first developed server-side scripting languages to be embedded into an HTML source document rather than calling an external file to process data.

The code is interpreted by a Web server with a PHP processor module which generates the resulting Web page. It also has evolved to include a common-line interface capability and can be used in stand-alone graphical applications. PHP can be deployed on most Web servers and also as a standalone shell on almost every operating system and platform free of charge.

Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic Webpage content or dynamic images used on Websites or elsewhere. It can also be used for command-line scripting and client-side biographical user interface (GUI) applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems (RDBMS). It is available free of charge and the PHP group provides the complete source code for users to build, customize and extend for their own use.

Instead of lots of commands to output HTML, PHP pages contain HTML with embedded code that does "something". The PHP code is enclosed in special ***start and end processing instructions <?php and ?>*** that allows you to jump into and out of "PHP mode".

Features of PHP

- Cost is low
- PHP is an open source software
- PHP is easy to learn
- PHP is embedded within HTML

Advantages of PHP Programming

PHP programming which had a humble beginning has now turned to be a major player in software development. It was successful in a humble beginning has now turned to be a major player in software development. It was successful in launching fast and simple scripting language that could work seamlessly with many other components.

The execution speed of a web application is very important as it cannot afford to slow down the rest of the machine, either. PHP integrates well with other software, especially under UNIX. It can be very fast, because it is a thin wrapper around many operating system calls.

PHP offers various security levels which can be set in the .ini file to defend the system from attacks of users, both as programmers and as surfers. Another important advantage of PHP is its simplicity. Even HTML coders can integrate PHP programming straight into their pages.

Further Advantages of PHP Programming include:

- Ease of writing interface to other libraries
- Several HTTP server interface.
- Large amount of database interface like MYSQL, MS SQL, Oracle, Infomix and Postgre SQL etc.
- PHP programming is similar to C/Javascript and java

Hyper Text Mark Up Language

An HTML file is a text file containing small markup tags. These tags tell the web browser how to display the page. An HTML file must have an htm or html file extension. An HTML file can be created by using a simple text editor. HTML documents are text files made up of HTML elements e.g.: <html>, <body>. HTML elements are defined using HTML tags.

HTML tags are used to markup HTML elements. The two characters surround HTML tags <and>. The surrounding characters are called angle brackets. HTML tags normally come in pairs like and . The first tag in a pair is the start tag: the second tag is the end

tag. The text between the start and the end tag is the element content. HTML tags are not case sensitive; <b> means the same as <B>.

MySQL

Relational database systems are the most important database systems used in the software industry today. One of the most outstanding systems is MySQL.

The important aspects of SQL Server are:

- MySQL is easy to use.
- Embedded database library.
- Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.

MySQL is an open-source relational database management system (RDBMS); in July 2013, it was the world's second most widely used RDBMS, and the most widely used open-source client–server model RDBMS. It is named after co-founder Michael Widenius's daughter, MySQL acronym stands for Structured Query Language. The MYSQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress , phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter,Flickr, and YouTube.

On all platforms except Windows, MySQL ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or install MySQL workbench via a separate download. Many third party GUI tools are also available.

MySQL is offered fewer than two different editions: the open source MySQL Community Server and the proprietary Enterprise Server. MySQL Enterprise Server is differentiated by a series of proprietary extensions which install as server plugins, but otherwise shares the version numbering system and is built from the same code base.

MySQL Features

- Localized to many languages
- Powerful security system.
- Master/Slave replication for high availability.
- Per table choice of backend(with/without transaction)
- Can be executed with user definition function(UDF)
- Embedded engine(on client server)
- Open source

SYSTEM DESIGN

4.1 SYSTEM DESIGN

The purpose of System Design is to create a technical solution that satisfies the functional requirements for the system. At this point in the project lifecycle there should be a Functional Specification, written primarily in business terminology, containing a complete description of the operational needs of the various organizational entities that will use the new system. The challenge is to translate all of this information into Technical Specifications that accurately describe the design of the system, and that can be used as input to System Construction.

The Functional Specification produced during System Requirements Analysis is transformed into a physical architecture. System components are distributed across the physical architecture, usable interfaces are designed and prototyped, and Technical Specifications are created for the Application Developers, enabling them to build and test the system. Many organizations look at System Design primarily as the preparation of the system component specifications; however, constructing the various system components is only one of a set of major steps in successfully building a system.

The preparation of the environment needed to build the system, the testing of the system, and the migration and preparation of the data that will ultimately be used by the system are equally important. In addition to designing the technical solution, System Design is the time to initiate focused planning efforts for both the testing and data preparation activities.

4.2 I/O DESIGN

The database tables are designed by using all the necessary fields in a compact and correct manner. Care has been taken to avoid redundant and duplicated data fields. The input as well as output forms were designed on the basis of the requirement study conducted in the system analysis phase.

User-friendly input forms have been designed with interactive dialogue. Data validation checks are done to avoid errors as much as possible. Using menus user can directly access input forms. Consistent format has been followed in developing forms. Help is available as and when required.

All input screens in the system are user friendly and are designed in such a way that even a layman can operate. The sizes of all the screens are standardized. Reports generated in this software give the finer accepts of the required information, which helps taking vital decisions.

4.3 DATABASE DESIGN

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the user.

In a database environment, common data are available in which several users can use. The concept behind a database is an integrated collection of data and provides a centralized access to the data from the program. It makes possible to treat data as a separate resource. While designing database, several objectives must be considered:

- Controlled Redundancy
- Data Independence
- More information at low cost
- Accuracy and Integrity
- Recovery from failure
- Privacy and Security.

4.4 TABLE DESIGN

Table design also deals with the design of physical database. A key field determines how the access is to be implemented. Suitable key fields are identified various tables and are coded approximately. The type, width and size of the fields are identified. Techniques of the

file organization are studied so that the most appropriate storage device can be selected and n different database design can be achieved. Specifying the data between them identifies the interface between the system databases and other databases. The designing of the tables in the database is done according to the rules specified for databases as described above. In the proposed project, tables are used. Insertion and retrieval of the values are easy by designing the database in this way.

Login Table

Column name	Data type	constraints
Login _Id	Int	Primary key
Login _ Username	Varchar(50)	Not Null
Login _ password	Varchar(50)	Not Null
Login_ type	Varchar(50)	Not Null

Parent Registration Table

Column name	Data type	constraints
User_ Id	Int	Primary key
User_ name	Varchar(50)	Not Null
User_ email	Varchar(50)	Not Null
User_ contact	Int	Not Null
User_ address	Varchar(50)	Not Null
User_ photo	Varchar(50)	Not Null

Doctor Table

Column name	Data type	constraints
Doctor_ Id	Int	Primary key
Doctor_ regid	Varchar(50)	Not Null
Doctor _ name	Varchar(50)	Not Null
Doctor_ email	Varchar(50)	Not Null
Doctor_ password	Varchar(50)	Not Null
Doctor_ contact	Varchar(50)	Not Null
Doctor_ experience	Varchar(50)	Not Null
Doctor_ working	Varchar(50)	Not Null
Doctor_ description	Varchar(50)	Not Null
Doctor_ address	Varchar(50)	Not Null
Doctor_ image	Varchar(50)	Not Null

Parent QueriesTable

Column name	Data type	constraints
Uq_ Id	Int (11)	Primary key
User email	Varchar(50)	Not Null
Doctor _ email	Varchar(50)	Not Null
User_ query	Varchar(50)	Not Null
User_ reply	Varchar(50)	Not Null
Created_ on	Varchar(50)	Not Null
Updated_on	Varchar(50)	Not Null

Speaker Booking Table

Column name	Data type	constraints
Booking_ id	Int	Primary key
Speaker_ id	Int	Foreign Key
Customer_ name	Varchar(50)	Not Null
Customer_ email	Varchar(50)	Not Null
Customer_ phone	Varchar(50)	Not Null
Customer_ address	Varchar(50)	Not Null
Required_ place	Varchar(50)	Not Null
Required_ time	Varchar(50)	Not Null
Required_ date	Varchar(50)	Not Null
Inform _ later	Int	Not Null
Message	Varchar(50)	Not Null
Status	Int	Not Null

Vaccination Table

Column name	Data type	constraints
V_ Id	Int (5)	Primary key
V_ name	Varchar(50)	Not Null
V_ cart	Varchar(50)	Not Null
V_ des	Varchar(50)	Not Null

4.5 DATA FLOW DIAGRAM

The data flow diagram (DFD) is one of the most important tools used by system analysis. It is used to define the flow of the system and its resources such as information. Data Flow Diagrams are a way of expressing system requirements in a graphical manner. DFD represents one of the most ingenious tools used for structured analysis Data flow diagrams are made up of a number symbols, which represent system components. The four basic symbols are used to construct DFD. They are symbols that represent data source, data flows, data transformations and data storage.

The symbols are:

- Square
- Circles
- Arrow
- Open Rectangle

A "square" defines a source (origination) or destination of a system data. A "circles" represents a process that transforms incoming data flows into outgoing data flows. An "arrow" identifies the data flow in motion. It is pipeline through which information is flown like the rectangle in the flowchart. An "open rectangle" is a data store.

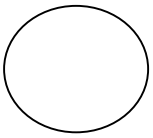
Six rules for construct a Data Flow Diagram:

- Arrows should not cross each other.
- Square, circles and file must be name.
- Decomposed data flow squares and circles can have same names.
- Choose meaningful names for data flow.
- Draw all data flows around the outside of the diagram.

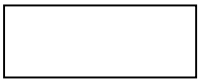
Objective of dataflow diagrams can be summarized as follows:

- To graphically document system boundaries.
- To show movement of data between the system and its environment.
- To provide hierarchical functional breakdown.
- To document intersystem information flows.
- To aid communication.

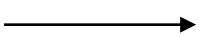
Basic symbols:



Process



External Entity



Data Flow

Data Store

Process:

A process transforms incoming data flow into outgoing data flow.

External Entity:

External entities are objects outside the system, with which the system communicates. External entities are sources and destinations of the system's inputs and output.

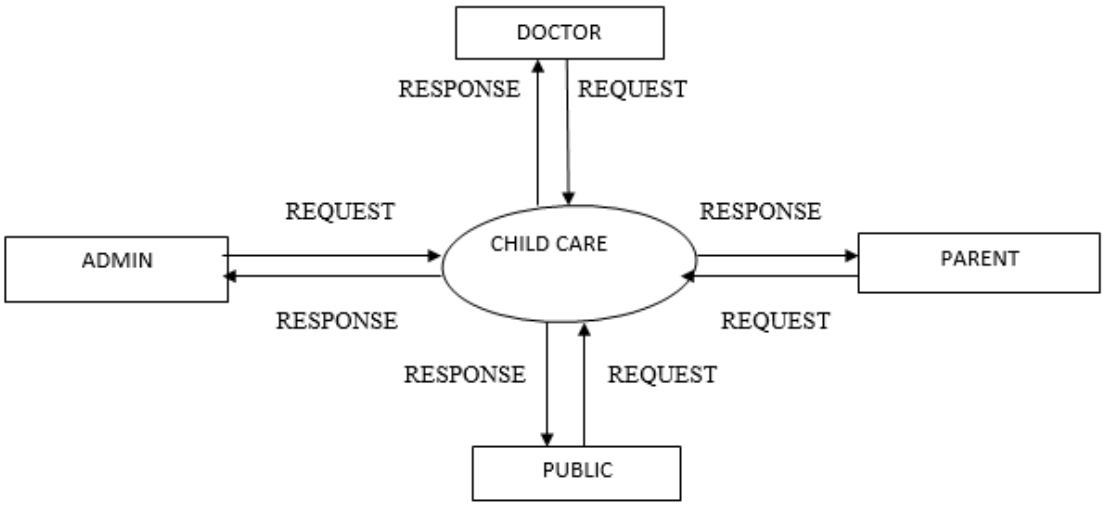
Data Flow:

Data flows are pipelines through which packets of information flow. Label the arrows with the name of the data that moves through it.

Data Store:

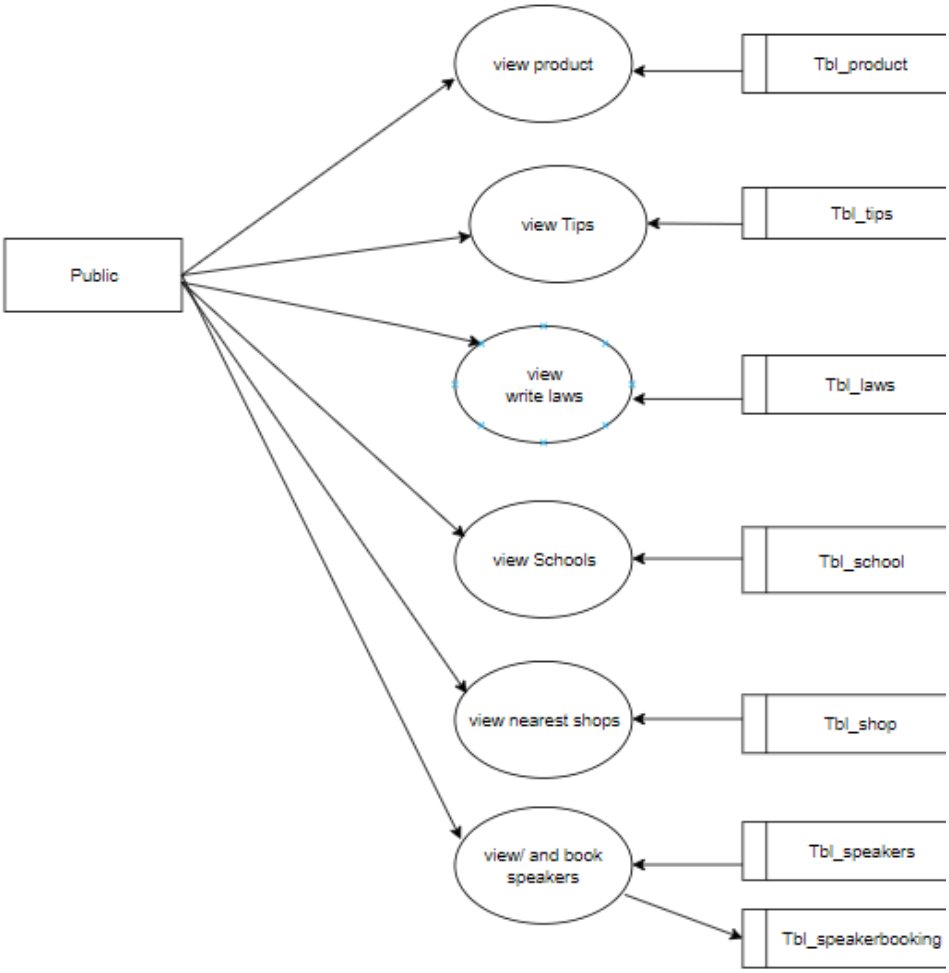
Data stores are repositories of data in the system. They are sometimes also referred to as files.

Level 0

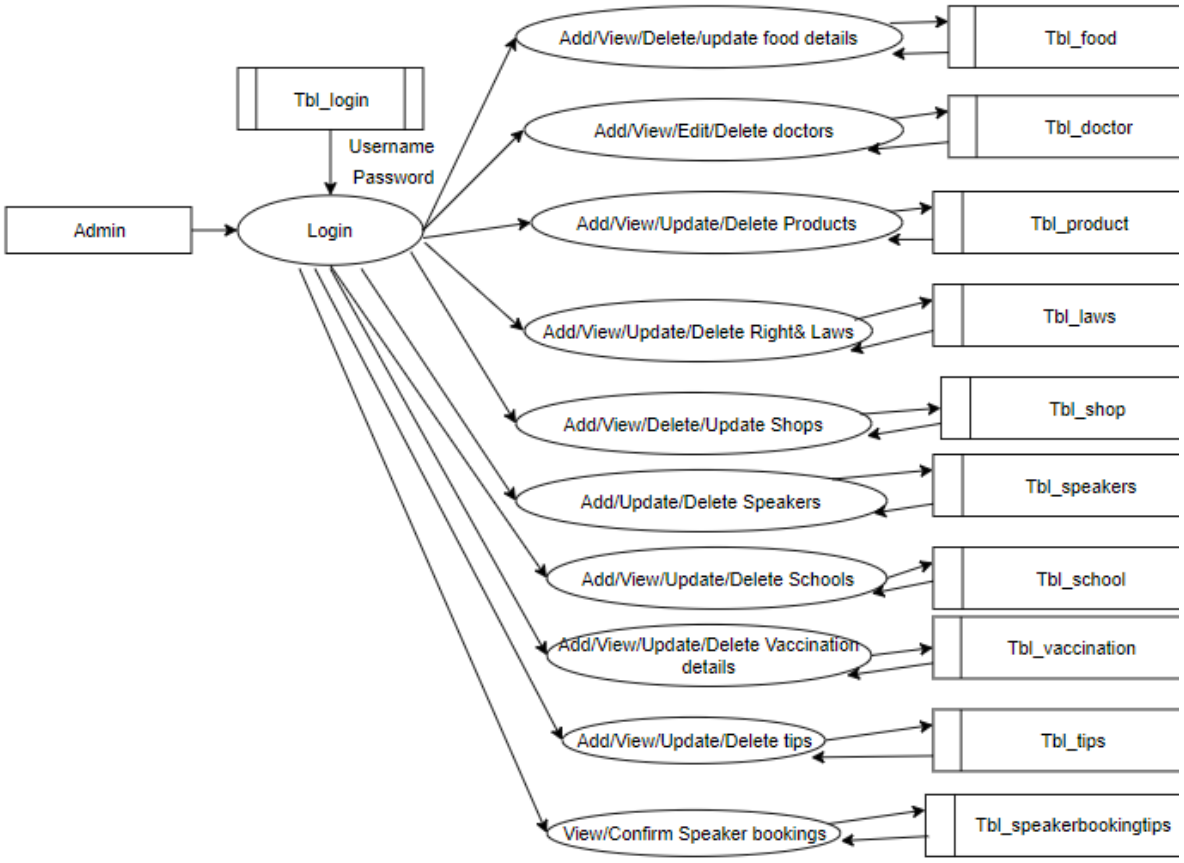

DOCTOR
RESPONSE
REQUEST
REQUEST
RESPONSE
ADMIN
CHILD CARE
PARENT
RESPONSE
REQUEST
RESPONSE
REQUEST
PUBLIC

Level 1:

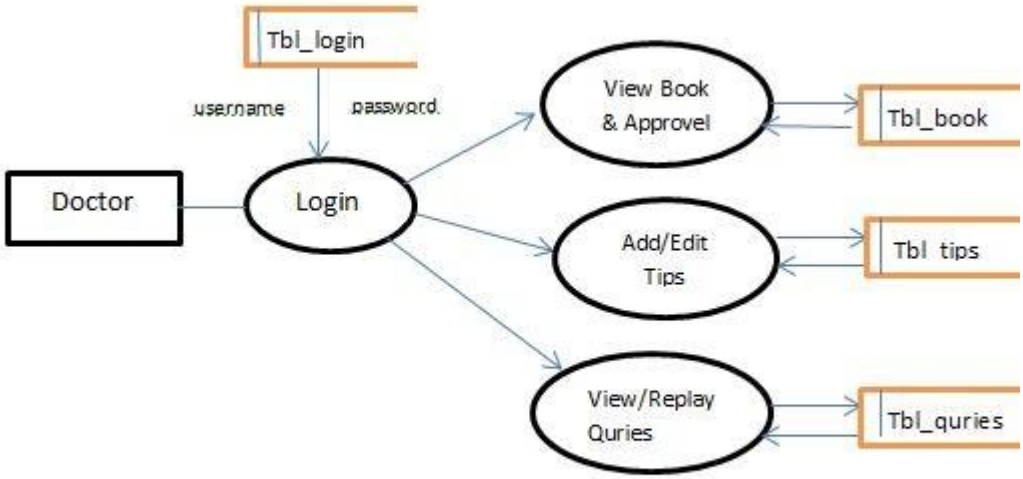
Public



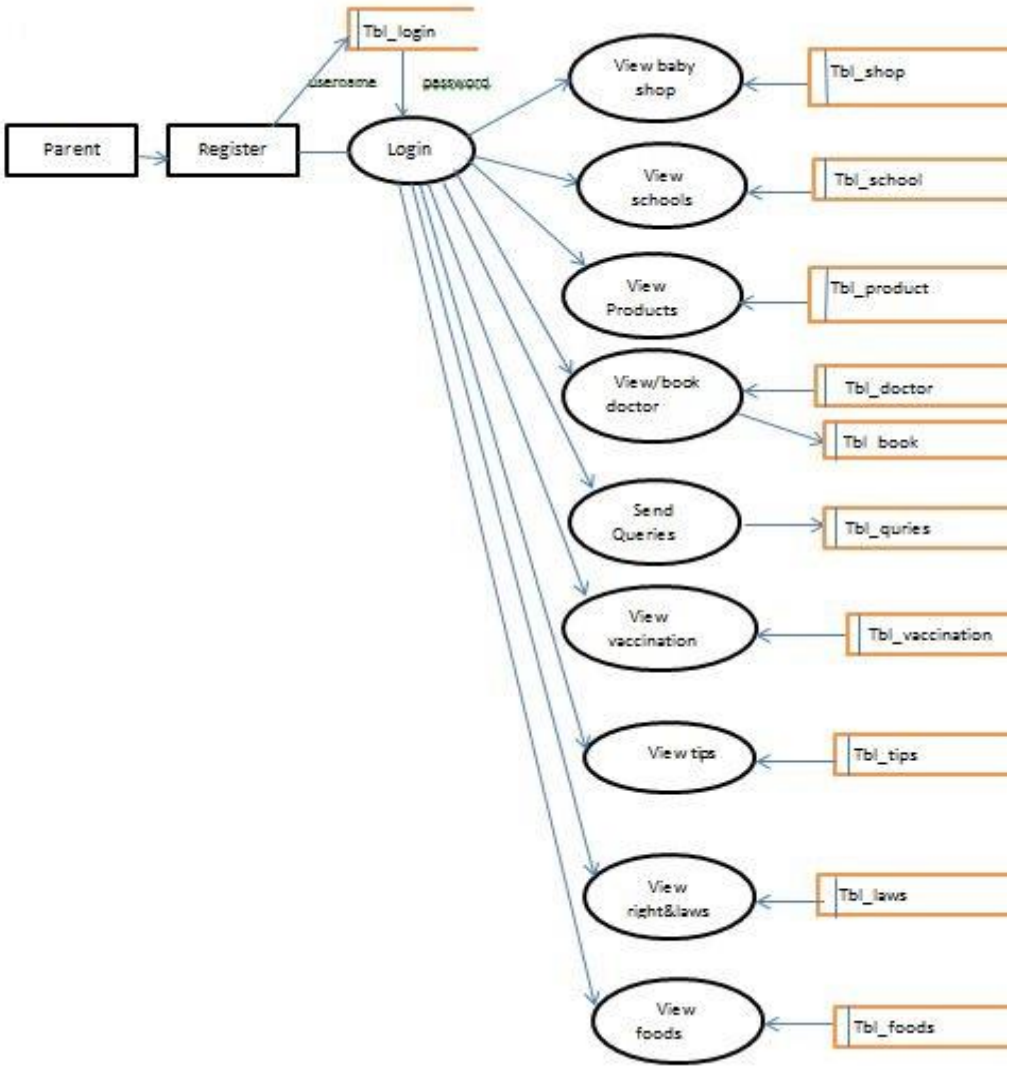
Admin


Admin
Login
Tbl_login
Username
Password
Add/View/Delete/update food details
Tbl_food
Add/View/Edit/Delete doctors
Tbl_doctor
Add/View/Update/Delete Products
Tbl_product
Add/View/Update/Delete Right& Laws
Tbl_laws
Add/View/Delete/Update Shops
Tbl_shop
Add/Update/Delete Speakers
Tbl_speakers
Add/View/Update/Delete Schools
Tbl_school
Add/View/Update/Delete Vaccination details
Tbl_vaccination
Add/View/Update/Delete tips
Tbl_tips
View/Confirm Speaker bookings
Tbl_speakerbookingtips

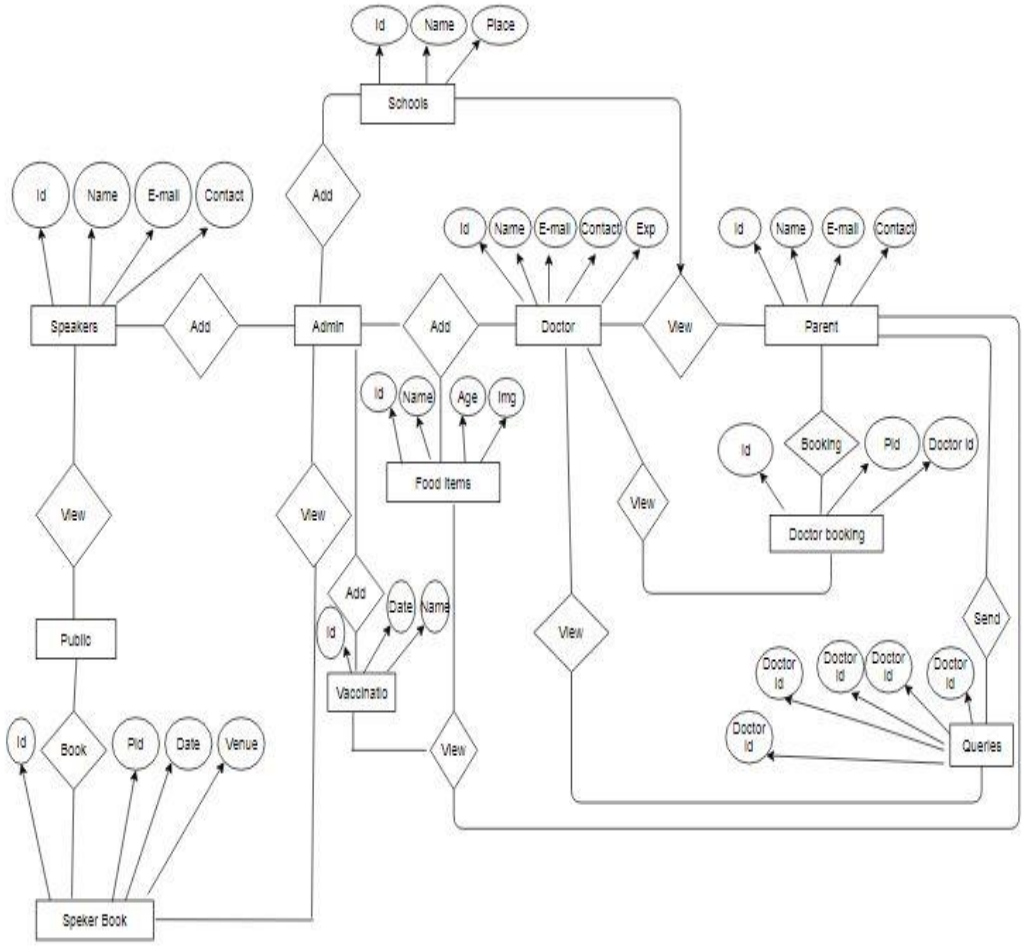
Doctor


Tbl_login
username
password
Doctor
Login
View Book
& Approvel
Tbl_book
Add/Edit
Tips
Tbl tips
View/Replay
Quries
Tbl_quries

Parent


Tbl_login
username
password
Parent
Register
Login
View baby shop
Tbl_shop
View schools
Tbl_school
View Products
Tbl_product
View/book doctor
Tbl_doctor
Tbl_book
Send Queries
Tbl_quries
View vaccination
Tbl_vaccination
View tips
Tbl_tips
View right&laws
Tbl_laws
View foods
Tbl_foods

4.6 ER-Diagram



4.7 Gantt Chart

I have planned all the tasks that must be carried out in order to accomplish the goal like Requirement gathering, Analysis, Designed, coding, testing, and implementation.

Time → / Task ↓	Week 1	Week 2	Week 3	Week 4	Week 5
Requirement Gathering					
Analysis					
Design					
Coding					
Testing					
Implementation					

4.8 PERT Chart

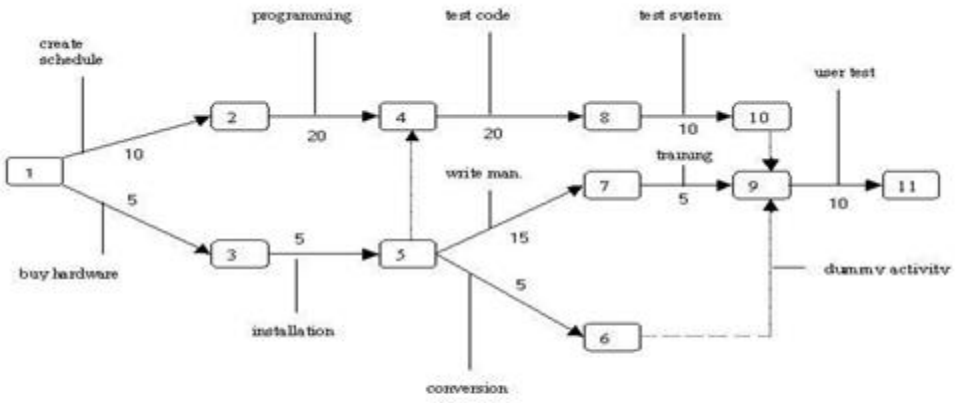


Fig. 1: PERT Chart

* Numbered rectangles are nodes and represent events or milestones.
* Directional arrows represent dependent tasks that must be completed sequentially.
* Diverging arrow directions (e.g. 1-2 & 1-3) indicate possibly concurrent tasks
* Dotted lines indicate dependent tasks that do not require resources.

CODING

Public Controller

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class publiccontrol extends CI_Controller
 {
  function __construct()
  {
   parent::__construct();
      $this->load->model('publicmodel');
  }
  public function index()
  {
   $data['rights']=$this->publicmodel->getrights();
   $data['speakers']=$this->publicmodel->getspeakers();
   $data['tips']=$this->publicmodel->gettips();
         $this->load->view('publicindex',$data);
  }
  public function signin()
  {
          $this->load->view('signin');
  }
  public function Registration()
  {
```

```
    $this->load->view('userregistration');
   }
   public function User_registration()
   {
    $name=$this->input->post('name');
    $email=$this->input->post('email');
    $contact=$this->input->post('contact');
    $address=$this->input->post('address');$password=$this->input->post('password');

$data=array('user_name'=>$name,'user_email'=>$email,'user_contact'=>$contact,'user_addre
ss'=>$address);

$data1=array('login_username'=>$email,'login_password'=>$password,'login_type'=>"user")
;
    $res=$this->publicmodel->registration($data,$data1);
    redirect('publiccontrol/signin');


   }
   public function Login()
   {
    $e=$this->input->post('email');
     $p=$this->input->post('password');

    $query=$this->publicmodel->login_check($e,$p);
   $data['type'] = $query['type'];
    $data['count'] = $query['count'];
    if($data['count']==1)
```

```
{
   $this->session->set_userdata('email',$e);
  if($data['type']=="doctor")
  {
    redirect('Pedeatrician_Doctor');

  }
  else if($data['type']=="user")
  {
redirect('User');
  }
}
else
{
  echo"<script>alert('Wrong username/Password')</script>";
  $this->signin();
}
}
public function Speaker_Request($id)
{
$data['id']=$id;
$this->load->view('speakerrequest',$data);

}
public function Give_Your_Complaints()
{
```

```
    $this->load->view('complaints');
   }
   public function Complaints_Giving()
   {
    $name=$_POST['name'];
   $email=$_POST['email'];
   $complaint=$_POST['complaint'];
   $curdate=date('Y-m-d');

$data=array('complaint_name'=>$name,'complaint_email'=>$email,'complaint_complaints'=
>$complaint,'complaint_date'=>$curdate);
  $this->publicmodel->complaint($data);
  echo"<script>alert('We will get back you soon')</script>";
  $this->index();


   }
   public function getbooking()
   {


$speakerid=$_POST['id'];
$name=$_POST['name'];
$email=$_POST['email'];
$contact=$_POST['contact'];
$address=$_POST['address'];
$place=$_POST['place'];
$time=$_POST['time'];
```

```
$rdate=$_POST['rdate'];

$inform=$_POST['inform'];

$message=$_POST['message'];

if($inform == "on")

{

  $inform=1;



}

else

{

  $inform=0;

}

$data=array('speakerid'=>$speakerid,'customer_name'=>$name,'customer_email'=>$email,'customer_phone'=>$contact,'customer_address'=>$address,'required_place'=>$place,'required_time'=>$time,'required_date'=>$rdate,'inform_later'=>$inform,'message'=>$message);

$result=$this->publicmodel->booking($data);

echo"<script>alert('Your Request has been sent.You will get an sms on confirmation of your request..Stay connected!')</script>";

$this->index();

   }



public function Schools_Near_You()

{

  $data['school']=$this->publicmodel->viewschools();

  $this->load->view('viewschools',$data);
```

```
}
public function Search_School()
{


}
public function Baby_Shop()
{
  $this->load->model('adminmodel');
$data['shops']=$this->adminmodel->viewshops();
  $this->load->view('viewbabyshops',$data);
}
public function Baby_Products()
{
  $data['products']=$this->publicmodel->viewproducts();
  $this->load->view('viewbabyproducts',$data);
}
}


?>
```

Admin Controller

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class Admin extends CI_Controller
  {
    function __construct()
```

```
{
 parent::__construct();
    $this->load->model('adminmodel');
}
public function index()
{
        $this->load->view('admin/adminlogin');
 // redirect('publiccontrol/signin');
  // redirect(base_url()."home/index/".$in);
}
public function Admin_Login()
{
        $username=$this->input->post('email');
        $password=$this->input->post('password');
        $res=$this->adminmodel->login($username,$password);
        if($res==True)
        {
   $this->session->set_userdata('email',$username);
                                $this->loginhome();
        }
        else
        {
                echo"failed";
                $this->index();
        }
}
```

```
public function loginhome()
{

        $this->load->view('admin/adminhome');
}
public function Complaints()
{
        $data['complaints']=$this->adminmodel->complaints();
        $this->load->view('admin/viewcomplaints',$data);
}
public function Remove_complaint($id)
{
 $res=$this->adminmodel->removecomplaint($id);
 $this->complaints();
}
Public function New_Speakers()
{
        $this->load->view('admin/addspeakers');
}
public function Speakers_Booking($id)
{
 $data['booking']=$this->adminmodel->speakerbooking($id);
  $this->load->view('admin/speakerbooking',$data);
}
public function Respond_Booking($id,$phone)
{
```

```
    $data=array('status'=>1);
    $result=$this->adminmodel->confirmbooking($data,$id);
    if($result==true)
    {
      $message="Dear Client , Your Request has been approved .Child care team will contact you soon.Have a nice day :-)";
      $this->SendSms($phone,$message);
      redirect('Speakers');
    }

  }
  public function Add_Speakers()
  {

      $name=$this->input->post('name');
      $contact=$this->input->post('contact');
      $email=$this->input->post('email');
      $experience=$this->input->post('experience');
      $address=$this->input->post('Address');
      $description=$this->input->post('description');
       $img= $_FILES['image']['name'];
       $time = Time();
       $images = explode('.',$img);
       $imagename =$time.'.'.end($images);
       $config['upload_path']          = './assets/speakers/';
       $config['allowed_types']        = 'jpg|png|jpeg';
```

```
$config['file_name'] = $imagename;
$this->load->library('upload', $config);
$this->upload->initialize($config);

if ( ! $this->upload->do_upload('image'))
   {
  echo"<script>alert('Error uploading image')</script>";

  $this->load->view('admin/addspeakers');
   }
   else
   {

$data=array('speaker_description'=>$description,'speaker_name'=>$name,'speaker_image'=>
$imagename,'speaker_email'=>$email,'speaker_address'=>$address,'speaker_experience'=>$
experience,'speaker_contact'=>$contact);

   $result=$this->adminmodel->addspeaker($data);
   if($result==True)
   {
     $this->Speakers();

   }
   else
   {
     echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
```

```
            $this->New_Speakers();
        }
        }
}
public function Speakers()
{
        $data['speakers']=$this->adminmodel->getspeakers();
        $this->load->view('admin/viewspeakers',$data);
}
public function Remove_speaker($id)
{
 $res=$this->adminmodel->removespeaker($id);
 $this->Speakers();
}

public function Edit_speaker($id)
{
  $data['speakers']=$this->adminmodel->getspeakersbyid($id);
$this->load->view('admin/editspeakers',$data);
}
public function Edit_Speakers()
{
 $id=$this->input->post('id');
 $name=$this->input->post('name');
    $contact=$this->input->post('contact');
    $email=$this->input->post('email');
```

```
$experience=$this->input->post('experience');
$address=$this->input->post('Address');
$description=$this->input->post('description');
 $img= $_FILES['image']['name'];
 if($img=="")
 {

$data=array('speaker_description'=>$description,'speaker_name'=>$name,'speaker_email'=>
$email,'speaker_address'=>$address,'speaker_experience'=>$experience,'speaker_contact'=>
$contact);


   $result=$this->adminmodel->editspeaker($data,$id);
   if($result==True)
   {
     $this->Speakers();


   }
   else
   {
    echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
    $this->New_Speakers();
   }


}
else
{
  $time = Time();
```

```
$images = explode('.',$img);
$imagename =$time.'.'.end($images);
$config['upload_path']          = './assets/speakers/';
$config['allowed_types']        = 'jpg|png|jpeg';

$config['file_name'] = $imagename;
$this->load->library('upload', $config);
$this->upload->initialize($config);

if ( ! $this->upload->do_upload('image'))
   {
 echo"<script>alert('Error uploading image')</script>";

 $this->load->view('admin/addspeakers');
   }
   else
   {

$data=array('speaker_description'=>$description,'speaker_name'=>$name,'speaker_image'=>
$imagename,'speaker_email'=>$email,'speaker_address'=>$address,'speaker_experience'=>$
experience,'speaker_contact'=>$contact);

   $result=$this->adminmodel->editspeaker($data,$id);
   if($result==True)
   {
     $this->Speakers();
```

```
        }
        else
        {
          echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
          $this->New_Speakers();
        }
        }
      }

}
 public function New_Doctor()
{
 $this->load->view('admin/addpedeatricians');
}
public function Doctors()
{
 $data['doctors']=$this->adminmodel->getdoctors();
 $this->load->view('admin/viewdoctors',$data);
}
public function Add_Doctor()
{

    $name=$this->input->post('name');
    $contact=$this->input->post('contact');
    $email=$this->input->post('email');
    $experience=$this->input->post('experience');
```

```
$address=$this->input->post('Address');
$description=$this->input->post('description');
 $place=$this->input->post('place');
  $regid=$this->input->post('regid');
  $psd="CHC".rand(1000,9000);
$img= $_FILES['image']['name'];
$time = Time();
$images = explode('.',$img);
$imagename =$time.'.'.end($images);
$config['upload_path']          = './assets/doctor/';
$config['allowed_types']        = 'jpg|png|jpeg';

$config['file_name'] = $imagename;
$this->load->library('upload', $config);
$this->upload->initialize($config);

if ( ! $this->upload->do_upload('image'))
  {
 echo"<script>alert('Error uploading image')</script>";

 $this->New_Doctor();
  }
  else
  {

$data=array('doctor_description'=>$description,'doctor_name'=>$name,'doctor_image'=>$im
```

```
agename,'doctor_email'=>$email,'doctor_address'=>$address,'doctor_experience'=>$experie
nce,'doctor_contact'=>$contact,'doctor_working'=>$place,'doctor_regid'=>$regid,'doctor_pas
sword'=>  $psd);


        $result=$this->adminmodel->adddoctor($data);
        if($result==True)
        {
          $this->Doctors();
          //SMS or EMAIL


        }
        else
        {
          echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
          $this->New_Doctor();
        }
        }



  }
  public function Remove_doctor($id)
  {
   $res=$this->adminmodel->removedoctor($id);
   $this->Doctors();
  }
   public function Edit_doctor($id)
```

```
 {
   $data['doctors']=$this->adminmodel->getdoctorbyid($id);
$this->load->view('admin/editdoctor',$data);
 }

public function Edit_Doctors()
{
  $id=$this->input->post('id');
  $name=$this->input->post('name');
     $contact=$this->input->post('contact');
     $email=$this->input->post('email');
     $experience=$this->input->post('experience');
     $address=$this->input->post('Address');
     $description=$this->input->post('description');
       $place=$this->input->post('place');
        $regid=$this->input->post('regid');
        $psd="CHC".rand(1000,9000);
      $img= $_FILES['image']['name'];
      if($img=="")
      {

$data=array('doctor_description'=>$description,'doctor_name'=>$name,'doctor_email'=>$em
ail,'doctor_address'=>$address,'doctor_experience'=>$experience,'doctor_contact'=>$contact,
'doctor_working'=>$place,'doctor_regid'=>$regid);


        $result=$this->adminmodel->editdoctor($data,$id);
        if($result==True)
```

```
    {
      $this->Doctors();

    }
    else
    {
     echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
      $this->Doctors();
    }
  }
  else
  {
   $time = Time();
  $images = explode('.',$img);
  $imagename =$time.'.'.end($images);
  $config['upload_path']          = './assets/doctor/';
  $config['allowed_types']        = 'jpg|png|jpeg';

 $config['file_name'] = $imagename;
 $this->load->library('upload', $config);
 $this->upload->initialize($config);

 if ( ! $this->upload->do_upload('image'))
    {
   echo"<script>alert('Error uploading image')</script>";
```

```
        $this->New_Doctor();
          }
          else
          {

$data=array('doctor_description'=>$description,'doctor_name'=>$name,'doctor_image'=>$im
agename,'doctor_email'=>$email,'doctor_address'=>$address,'doctor_experience'=>$experie
nce,'doctor_contact'=>$contact,'doctor_working'=>$place,'doctor_regid'=>$regid);


          $result=$this->adminmodel->editdoctor($data,$id);
          if($result==True)
          {
            $this->Doctors();


          }
          else
          {
            echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
            $this->Doctors();
          }
          }
        }


  }


public function Schools()
{
```

```
  $data['schools']=$this->adminmodel->getschool();
    $this->load->view('admin/viewschools',$data);
}
public function Remove_school($id)
{
  $res=$this->adminmodel->removeschool($id);
    $this->Schools();
}
public function Edit_school($id)
{
$data['schools']=$this->adminmodel->getschoolbyid($id);
  $this->load->view('admin/editschool',$data);
}
public function New_school()
{
  $this->load->view('admin/addschool');
}
public function Add_school()
{
        $map=$this->input->post('map');
        $name=$this->input->post('name');
        $contact=$this->input->post('contact');
        $email=$this->input->post('email');
        $type=$this->input->post('type');
        $address=$this->input->post('address');
       $img= $_FILES['image']['name'];
```

```
$time = Time();
 $images = explode('.',$img);
 $imagename =$time.'.'.end($images);
 $config['upload_path']          = './assets/schools/';
 $config['allowed_types']        = 'jpg|png|jpeg';

$config['file_name'] = $imagename;
$this->load->library('upload', $config);
$this->upload->initialize($config);

if ( ! $this->upload->do_upload('image'))
   {
  echo"<script>alert('Error uploading image')</script>";

  $this->New_school();
   }
   else
   {

$data=array('school_name'=>$name,'school_image'=>$imagename,'school_email'=>$email,'school_address'=>$address,'school_contact'=>$contact,'school_type'=>$type,'school_maplocation'=>$map);

   $result=$this->adminmodel->addschool($data);
   if($result==True)
   {
      $this->Schools();
```

```
            //SMS or EMAIL


          }
          else
          {
            echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
             $this->New_school();
          }
          }



}

public function Edit_schools()
{
  $id=$this->input->post('id');
  $map=$this->input->post('map');
       $name=$this->input->post('name');
       $contact=$this->input->post('contact');
       $email=$this->input->post('email');
       $type=$this->input->post('type');
       $address=$this->input->post('address');
      $img= $_FILES['image']['name'];
      if($img=="")
      {
```

```
$data=array('school_name'=>$name,'school_email'=>$email,'school_address'=>$address,'sch
ool_contact'=>$contact,'school_type'=>$type,'school_maplocation'=>$map);


        $result=$this->adminmodel->editschool($data,$id);
        if($result==True)
        {
            $this->Schools();



        }
        else
        {
          echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
              $this->Schools();
        }
      }
      else
      {
       $time = Time();
        $images = explode('.',$img);
        $imagename =$time.'.'.end($images);
        $config['upload_path']          = './assets/schools/';
        $config['allowed_types']        = 'jpg|png|jpeg';


       $config['file_name'] = $imagename;
```

```
$this->load->library('upload', $config);

$this->upload->initialize($config);


if ( ! $this->upload->do_upload('image'))

   {

  echo"<script>alert('Error uploading image')</script>";


   $this->Schools();

   }

   else

   {

$data=array('school_name'=>$name,'school_image'=>$imagename,'school_email'=>$email,'school_address'=>$address,'school_contact'=>$contact,'school_type'=>$type,'school_maplocation'=>$map);


   $result=$this->adminmodel->editschool($data,$id);

   if($result==True)

   {

        $this->Schools();


   }

   else

   {

     echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";

        $this->Schools();

   }
```

```
        }
      }

}
public function Right_Laws()
{
  $data['rights']=$this->adminmodel->viewrights();
  $this->load->view('admin/viewrightlaws',$data);
}
public function New_Right_Laws(){
  $this->load->view('admin/addrightslaws');
}
public function Add_Right_Laws(){
   $curdate=date('d M Y');
   $caption=$this->input->post('caption');
   $description=$this->input->post('description');

$data=array('rights_caption'=>$caption,'right_description'=>$description,'right_date'=>$curd
ate);
   $res=$this->adminmodel->addright($data);
   if($res==True)
         {
              $this->Right_Laws();


         }
         else
```

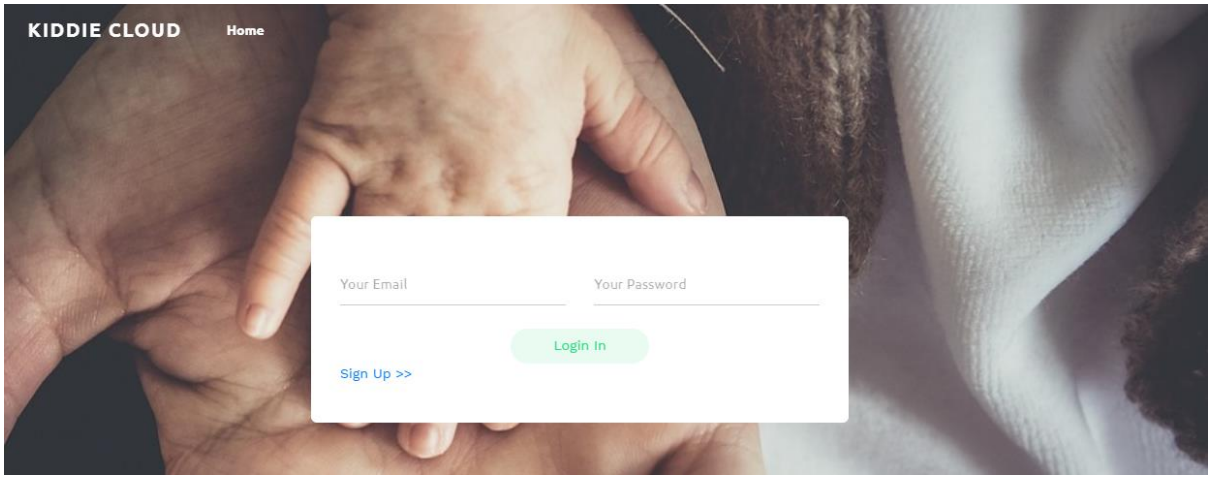
```
        {
          echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
              $this->New_Right_Laws();
        }


}
public function Remove_rights($id)
{
$res=$this->adminmodel->removeright($id);
   $this->Right_Laws();
}
public function Edit_rights($id)
{
$data['rights']=$this->adminmodel->getrightbyid($id);
$this->load->view('admin/editright',$data);
}
public function Edit_right()
{
$id=$this->input->post('id');
$caption=$this->input->post('caption');
  $description=$this->input->post('description');
  $data=array('rights_caption'=>$caption,'right_description'=>$description);
  $res=$this->adminmodel->editright($data,$id);
  if($res==True)
        {
             $this->Right_Laws();
```

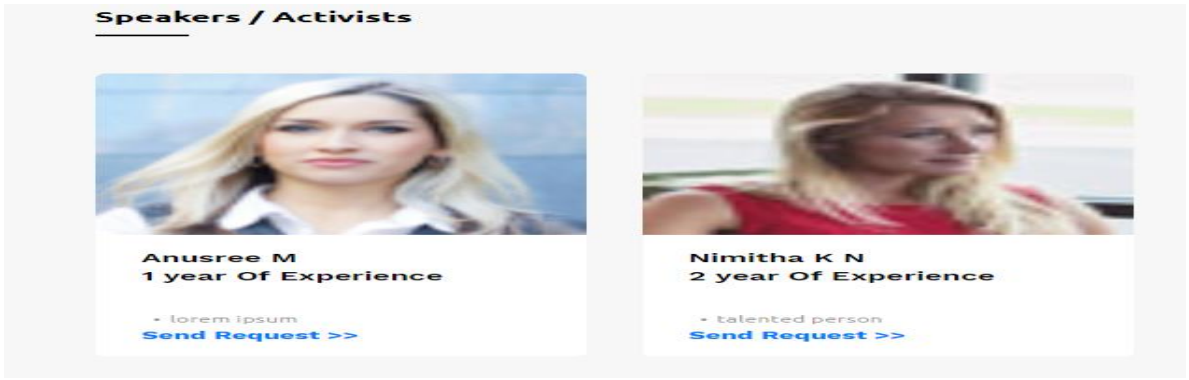
```
        }
        else
        {
          echo"<script>alert('Server error.Couldn't Upload your data!!')</script>";
            $this->Right_Laws();
        }
}
public function Show_products()
{
$data['products']=$this->adminmodel->viewproducts();
  $this->load->view('admin/viewproducts',$data);
}
public function New_product()
{
  $this->load->view('admin/addproduct');
}
?>
```

SCREEN SHOT

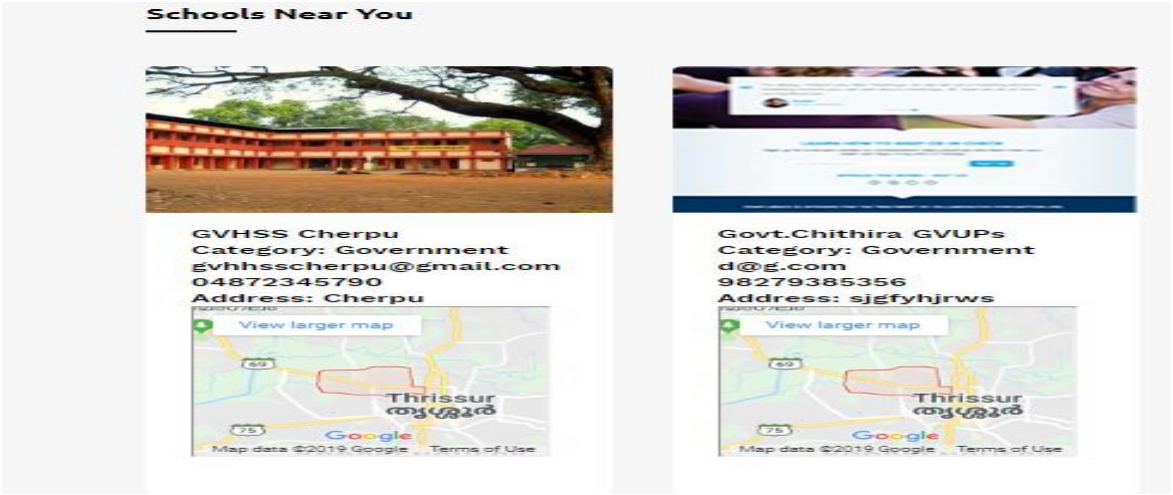
Login Form



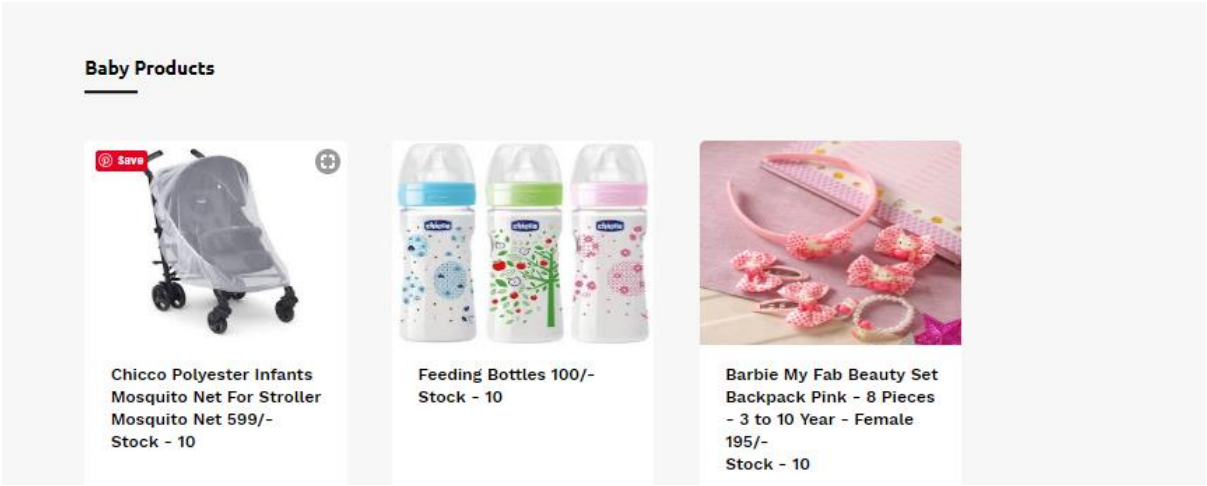
Speakers/Activists



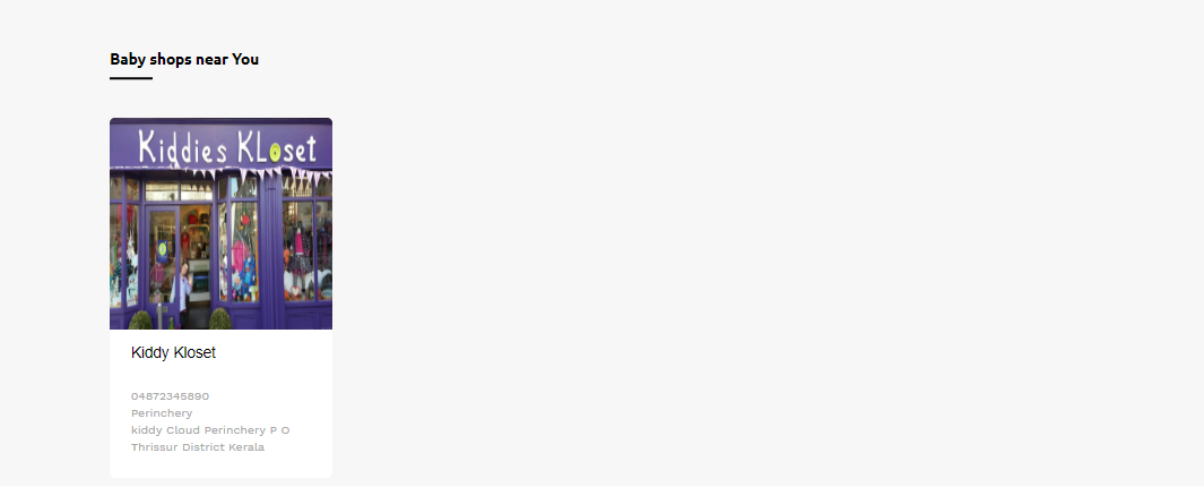
Schools



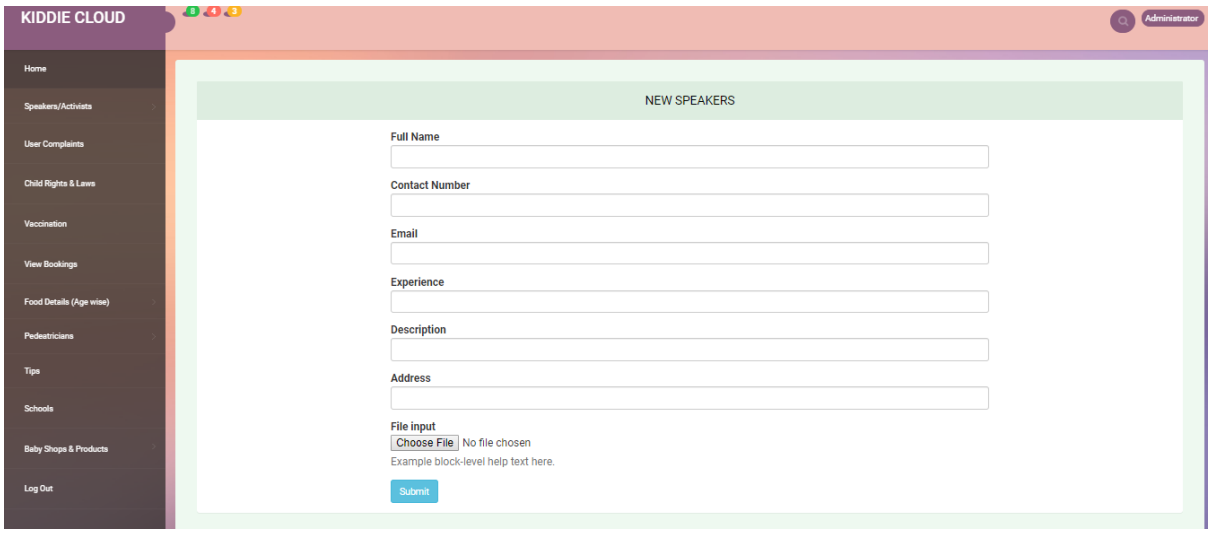
Baby Products



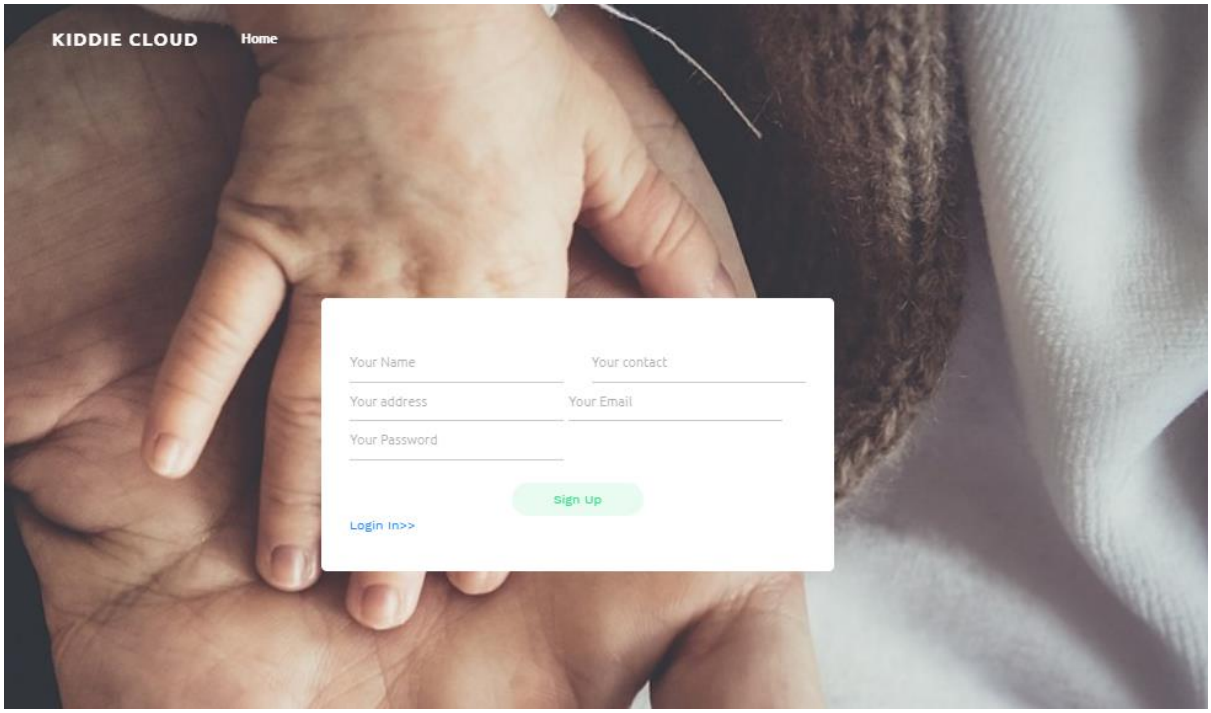
Shopes



Adding Speakers/Activists Form



Parent Registration Form

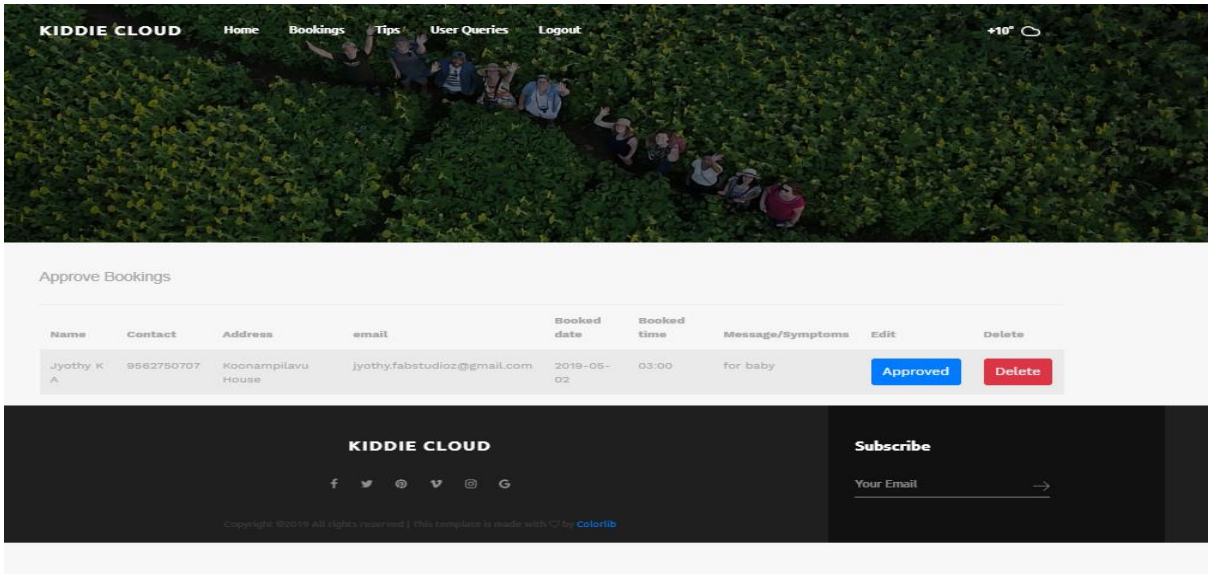


View Speaker Bookings

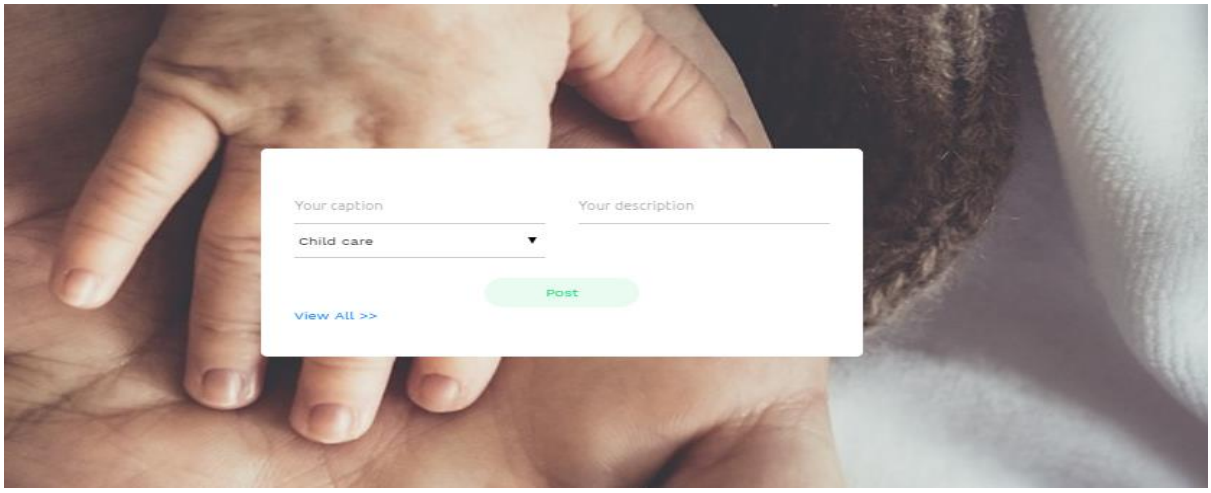
VIEW BOOKINGS

User Email	Doctor Email	Date	Time	Description	Action
shiya@gmail.com	g@gmail.com	2019-03-20	11:59	DGVNBCGFDVH	Approved
jyothy.fabstudioz@gmail.com	musthafa.fabstudioz@gmail.com	2019-05-02	03:00	for baby	Approved

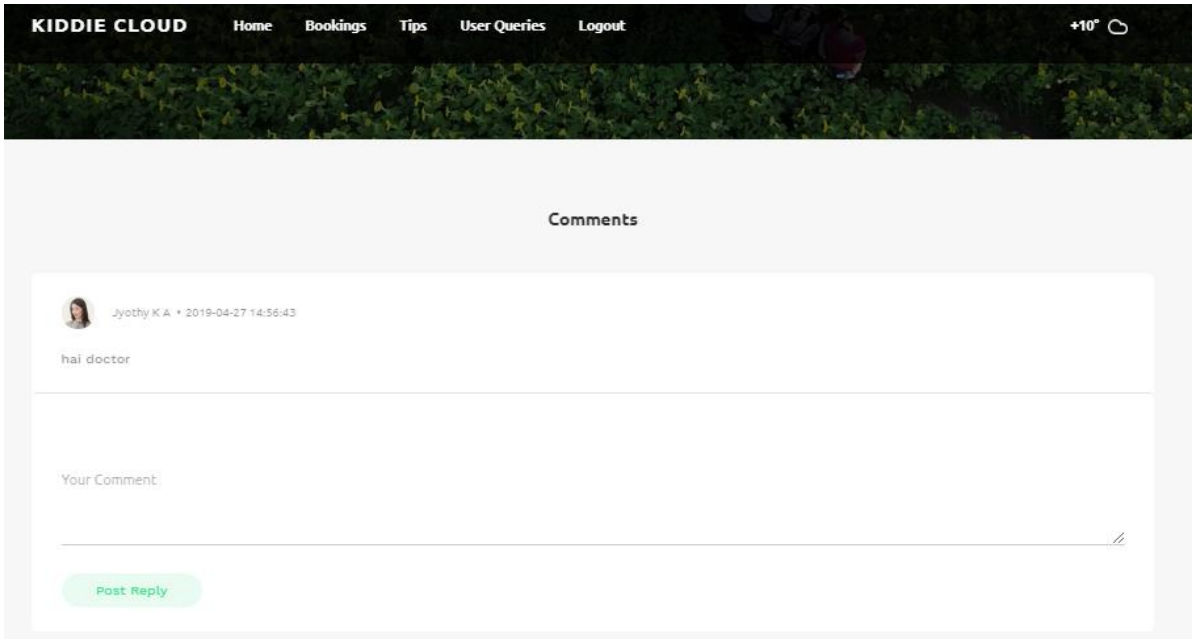
View Doctor Bookings



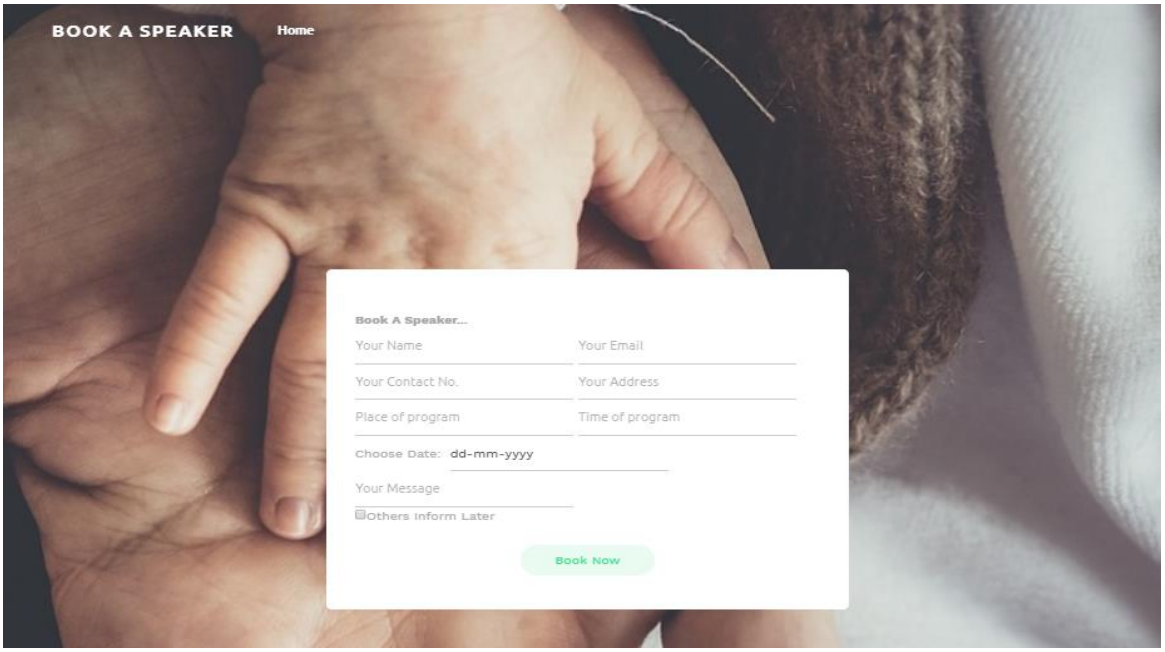
Add Tips Form



View Parent Queries



Speaker Booking Form



Adding Food Details Form

Adding Sports Event Form

SYSTEM TESTING

TESTING

Testing presents an interesting anomaly for the software engineering activities, the engineer attempts to build software from an abstract concept to a tangible product. Now comes testing. The engineer creates a series of test case that are initiated to "demolish" the software that has been build. Infect, testing is the one step in the software process that could be viewed (psychologically, at least) as destructive rather than constructive. Models of Testing:-There are different Models of testing. On the basis of testing methods there are two types of testing:

1. Black-box testing.

2. White-box testing

Black-box tests are used to demonstrate that software functions are operational, that input is properly accepted and output is correctly produced, and that integrity of external information is maintained. White-box tests are used to examine the procedural details. It checks the logical paths by test case. It can also checks the conditions, loops used in the software coding. It checks that loops are working correctly on defined boundary value.

White-Box Testing:

White-box testing some times called glass-box testing, is a test case design method that users the control structure of the procedural design to drive the test case. Always we are thinking that there is no necessary to execute or checks the loops and conditions. And so large number of errors is uncovered. With using white-box testing methods, we have checked that; All independent paths within a function have been executed at least once. All logical decisions on their true and false side. All loops working correctly at their boundary values and within their specified conditions. In our coding we test that all the loops works truly in each module. The one technique of white-box testing is basis path testing. It contains two parts, one is flow graph notation and the second is cyclometer complexity. In flow graph notation we are checking logical control of flow. By using cyclometer complexity we find complexity of our project structure.

Black-Box Testing:

Black-box testing focuses on the functional requirements of the software. That is black- box testing enables the software engineer to drive sets of input conditions that will fully exercise all functional Requirements for the program. Black-box testing is not an alternative to white-box testing techniques. Rather, it is a complementary approach that is likely to uncover a different class of errors than white-box methods. We use in our coding to find errors in the following categories:

- Incorrect or missing functions
- Interface errors
- Errors in database
- Performance errors
- Initialization and termination errors

Unit Testing:

The first level of testing is unit testing. This test focuses on each module separately assuming that it functions properly as a unit. Unit testing focuses verification efforts on the smallest unit of software design, the module. This is also known as "Module Testing". The modules are tested separately. This testing is carried out during programming stage itself. In this testing step each module is found to be working satisfactorily as regard to the expected output from the module. Developer and supervisor do the testing in order to trace out the bugs in the minimal part of the code. The goal is to test the internal logic of the module. Unit testing is typically done by the developers and not by end-users

Integration Testing:

Integration testing focuses on the design and the construction of the software architecture. The data can be lost across the interface or one module can pose an adverse effect on another. The sub functions when combined may not produce the major function. Integration testing is a systematic technique for the program structure, while at the same conducting test to uncover errors associated with the interface.

System Testing:

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved. For user acceptance testing, the system was given to the end user to use. The errors found are rectified.

IMPLEMENTATION AND CONCLUSION

IMPLEMENTATION OF SECURITY MECHANISMS

Information security is the protection of the information against unauthorized discloser, alteration or destruction .Database security is the protection of information that is maintained in a database. It is deals with ensuring that only "authorized person" can get access "authorized data". System used for processing sensitive information is prone to high security risk. Individual often tries to access unauthorized data for various reasons. Threats could be external or internal. In order to have good security User Name and Password authentication is given to the System so that only valid users can access the System. For the database point of view we make the constraint of the table field and apply locking system of the updating and editing the database records.

CONCLUSION

This is a web portal which mainly aims at parenting and child care. This system concentrates on children in the age group of 1 to 10. Their vaccination, food, diet and nutrition details where all will be available from this system. Parents can even book for pediatricians. This will be very helpful system for the parents to take care their children with proper guidence. They can sent their doubts to admin and will get replied. Child care may be defined as care for young children, provided by adults who are not their parents. Informal child care by relatives, nannies, or home care providers typically takes place in a home setting, while formal care by trained and untrained caregivers takes place in school or care center setting.

FUTURE EXTENSION AND BIBLIOGRAPHY

FUTURE EXTENSIONS

Kiddie Cloud System has as much future Scope as we think of. Some of the few point have come to light. They are as follows: -

1. Making it as an android app which can be integrated in faculties mobile.
2. We can include more an Online purchasing propery
3. We can provide kids care center.

BIBLIOGRAPHY

1. https://www.w3schools.com/

2. https://www.php.net/manual/en/intro-whatis.php.

3. https://stackoverflow.com/.

4. Learning php,mysql,javascript by Robin Nixon.