**Cake Api Document**

**What is Cake php ...?**

CakePHP is an open-source web, rapid development framework that makes building web applications simpler, faster and require less code.

**Benifits**

Api Url Stucture:- <http://192.168.0.128/i_gotcha/users/add.json>

1. [192.168.0.128](http://192.168.0.128/i_gotcha/users/add.json) => Host Name

2. [i\_gotcha](http://192.168.0.128/i_gotcha/users/add.json) => Project Name

3. [users](http://192.168.0.128/i_gotcha/users/add.json) => Controller Name

4. [add.json](http://192.168.0.128/i_gotcha/users/add.json) => Action Name

**Add User Functionality**

http://192.168.0.128/i\_gotcha/users/add.json

Request Api

{

"contactno":"111111",

"email":"t11111@gmail.com",

"fname": "11test",

"lname" :"11test",

"password":"1234567890",

"profile\_img" :"111\_18039.png"

}

Response =>

{

"users": {

"contactno": "111111",

"email": "t11111@gmail.com",

"fname": "11test",

"lname": "11test",

"profile\_img": "111\_18039.png",

"api\_key\_plain": "771052ec5581898081b3f41c5bbc5f1abfe82004",

"api\_key": "$2y$10$XqbkNiWHSecf1cNFpoyVPOVjhRMbSgAHTqrBz\/KPh.w.DYgjX5H3y",

"created": "2017-03-18T06:44:25",

"modified": "2017-03-18T06:44:25",

"id": 35,

"status": 1,

"message": "The user has been saved."

}

}

**Edit User** **Functionality**

http://192.168.0.128/i\_gotcha/users/edit.json

Request

{

"contactno":"111111",

"email":"t11111@gmail.com",

"fname": "11test",

"lname" :"11test",

"password":"1234567890",

"profile\_img" :"111\_18039.png",

"id" :"39",

}

Response

{

users = {

message = "The user has been updated.";

status = 1;

};

}

**Login User**

http://192.168.0.128/i\_gotcha/users/login.json

In header => username, api\_plain\_key

Request:

{

email = "t111kkk11@gmail.com";

password = 123456;

}

Filttering data with **User id & Contain**

link => <http://192.168.0.128/i_gotcha/users/view.json>

**Request**

{

"conditions": {"Users.id": 16},

"contain": [ "Feeds"],

"fields": [],

"get": "all",

"order": [],

"page": 1,

"limit": 10

}

**Response**

{

"**users**": [

{

"id": 16,

"face\_personid": "fsagsagsa",

"role\_type": 1,

"fname": "dsgdsg",

"lname": "dgdsvbg",

"gender": "Male",

"profile\_img": "dsgdsgdg",

"face\_img": "sgdsgg",

"email": "test@gmail.com",

"contactno": "123156498a",

"$2y$10$59brGRnEN.wubTlwnWl04.AU1cJ.S54zWhnmA9z4BUyLIkVBYsl9W",

"api\_key\_plain": "290cd3ae3da1f5aa6a56cfac1f86782e4109511b",

"**feeds**": [

{

"id": 60,

"user\_id": 16,

"image\_typeid": 0,

"place\_name": "test",

"latitude": "464",

"longitude": "5454",

"vote\_cnt": 0,

"status": true,

"isDelete": false,

"created": "2017-03-14T08:59:19",

"modified": "3\/14\/17 8:59 AM"

}

]

}

]

}

Filttering data with **Like Query**

**Request**

{

"conditions":{"Users.email LIKE":"%teast@gmail.com%"},

"contain": [ "Feeds"],

"fields": [],

"get": "all",

"order": [],

"page": 1,

"limit": 10

}

**Response**

{

"**users**": [

{

"id": 17,

"face\_personid": "fsagsagsa",

"role\_type": 1,

"fname": "dsgdsg",

"lname": "dgdsvbg",

"gender": "Male",

"profile\_img": "dsgdsgdg",

"face\_img": "sgdsgg",

"email": "teast@gmail.com",

"contactno": "123156498a",

"alert\_points": true,

"alert\_sound": true,

"alert\_recognization": true,

"user\_level": 0,

"points": 100,

"api\_key": "$2y$10$bVxzsC9pw\/LW4mFfYRKoD.xpO7Yn5tv\/z1RuweCJZ2iG2RZ5\/8E.e",

"api\_key\_plain": "c386ef46a97738dfe25094ce0ab59df5e7f5f74b",

"status": true,

"city": "",

"state": "",

"country": "",

"isBlocked": false,

"isDelete": false,

"created": "2017-03-15T13:53:40",

"modified": "3\/15\/17 1:53 PM",

"feeds": []

}

]

}

Filttering data with **Like Query & Limit display data**

**Request**

{

"conditions":[],

"contain": ["Feeds"],

"fields": ["Users.id","Users.fname"],

"get": "all",

"order": [],

"page": 1,

"limit": 10

}

**Response**

{

"**user**": [

{

"id": 4,

"fname": "test",

"feeds": []

},

{

"id": 8,

"fname": "dsgdsg",

"feeds": []

},

{

"id": 9,

"fname": "dsgdsg",

"feeds": []

},

{

"id": 10,

"fname": "test",

"feeds": []

},

{

"id": 11,

"fname": "test",

"feeds": []

}

]

}

Filltering data with **ASC & DESC**

**Request**

{

"conditions":[],

"contain":["Feeds"],

"fields":["Users.id","Users.fname"],

"select":[],

"get":"all",

"order":{"Users.id":"Desc"},

"page":1,

"limit":10

}

**Response** => Desc

{

"user": [

{

"id": 39,

"fname": "11test44444444444444",

"feeds": []

},

{

"id": 38,

"fname": "11test",

"feeds": []

},

{

"id": 37,

"fname": "11test",

"feeds": []

}

]

}

**Response** => ASC

{

"user": [

{

"id": 4,

"fname": "test",

"feeds": []

}

]

}

Filltering data with **OR Condition**

**Request**

{

"conditions": {

"or": {

"Users.id": 8,

"Users.api\_key\_plain": "6332b046e74e61d4a3df3341a47fe21351d4e54c"

}

},

"contain": ["Feeds"],

"fields": [],

"select": [],

"get": "all",

"order": [],

"page": 1,

"limit": 10

}

**Response**

{

"user": [

{

"id": 4,

"face\_personid": "dsgdshdsds",

"api\_key": "$2y$10$b5zI2k3til9w.X28UZW7AOZxCP.gv4oGg1a2p07wgalo6lpWAL2wO",

"api\_key\_plain": "6332b046e74e61d4a3df3341a47fe21351d4e54c",

"status": true,

"created": "2017-03-15T12:03:38",

"modified": "3\/15\/17 12:03 PM",

"feeds": []

},

{

"id": 8,

"face\_personid": "fsagsagsa",

"role\_type": 1,

"api\_key": "$2y$10$CxL2h81qCqIxK1Exf02FkOCADemSxZI8KWCbJy3oAJEsDU5vHsuDi",

"api\_key\_plain": "02f87442147610c5660c511464581df2f56c8c63",

"isBlocked": false,

"isDelete": false,

"created": "2017-03-15T12:26:11",

"modified": "3\/15\/17 12:26 PM",

"feeds": []

}

]

}

http://www.jqueryscript.net/animation/jQuery-Plugin-To-Add-CSS3-Scrolling-Effects-On-Your-Web-Page-Smoove.html

http://www.jqueryscript.net/animation/jQuery-Plugin-For-Stacked-Page-Scrolling-Effect-jQuery-Stacks.html

UBUNTU COMMAND

folder directory techno@techno-100:~$ cd /opt/lampp/htdocs/cakephp

cake bake bin/cake bake all tablenames

conroller name

techno@techno-100:~$ sudo su

[sudo] password for techno:

root@techno-100:/home/techno# /opt/lampp/lampp restart

or

sudo /opt/lampp/lampp start

API

make table with api\_key & api\_key\_plain

download cakephp & put in htdocs

config-> app.php change dbname,username,password

folder directory techno@techno-100:~$ cd /opt/lampp/htdocs/cakephp

cake bake bin/cake bake all tablenames

conroller name

in route.php

put $routes->extensions(['json', 'xml']); before

$routes->fallbacks(DashedRoute::class);

in appcontroller for load auth component in initialize function

$this->loadComponent('Auth', [

'authenticate' => [

'Basic' => [

'fields' => ['username' => 'email', 'password' => 'api\_key'],

'userModel' => 'Users'

]

],

'loginAction' => [

'controller' => 'Users',

'action' => 'login'

]

]);

in appcontroller(api side)

public function serialize($code = 500, $message = null, $data = array())

{

// $controller = strtolower($this->request->params['controller']);

$controller = $this->request->params['controller'];

$action = $this->request->params['action'];

$url = $controller.'/'.$action.'.json';

$data\_var = '';

$$controller = [];

if(!$message) $message = 'Error occured! Please try again.';

if($data) $$controller = $data;

// pr($$controller); exit();

$this->set(compact(['code', 'url', 'message', $controller]));

$this->set('\_serialize', ['code', 'url', 'message', $controller]);

}

public function saveImage($file, $key)

{

// GET Request data

// $res = $this->request->data;

//set allowed extensions

$arr\_ext = array('jpg', 'jpeg', 'gif', 'png', 'mp4', 'avi', 'm4v', '3gp', 'mov');

$imgArr = array();

if($file){

$folder = 'media\_image\_folder/new';//str\_replace(array('\_image\_folder'), '', $key);

$folder1 = 'media\_image\_folder/actual';

$name = $file['name'];

$tmp\_name = $file['tmp\_name'];

$type = $file['type'];

$arr = pathinfo($name);

$ext = $arr['extension'];

$newName = time() . "\_" . rand(000000, 999999);

if (in\_array($ext, $arr\_ext)) {

// $dest\_path = WWW\_ROOT."img/$folder/".$newName.'.'.$ext;

$dest\_path = WWW\_ROOT."uploads/$folder/".$newName.'.'.$ext;

$var = move\_uploaded\_file($tmp\_name, $dest\_path);

if($var) {

// copy($folder . '1.jpg', $folder . '2.jpg');

$dest\_path1 = WWW\_ROOT."uploads/$folder1/".$name;

copy($dest\_path, $dest\_path1);

$imgArr['orginal\_name'] = $name;

$imgArr['new\_name'] = $newName.'.'.$ext;

$imgArr['type'] = $type;

}

}

}

//print\_r($imgArr); exit();

return $imgArr;

}

for login

<http://www.codexworld.com/cakephp-tutorial-beginners/>

<http://www.codexworld.com/cakephp-tutorial-part-2-add-edit-delete-operations/>

<http://www.codexworld.com/cakephp-tutorial-part-3-working-with-elements-layout/>

Request

$this->request->params['controller'];

$this->request->params['action'];

$this->request->data

$this->request->params['pass'][0];

$this->request->params['pass'][1];

syntax for cake

post /get data give after form submit :

$params = $this->request->getData();

$params = $this->request->getData('name of field');

$this->request->data['uesrname']

is post

: if ($this->request->is('post')) {

request is

if ($this->request->is(['patch', 'post', 'put'])) {

Insert post /get data into db

$registration = $this->Registrations->newEntity(); // create blanck obj

$registration = $this->Registrations->patchEntity($registration, $this->request->getData());

$this->Registrations->save($registration) // save method for save

if ($this->Registrations->save($registration)) {

$this->Flash->success(\_\_('The registration has been saved.'));

return $this->redirect(['action' => 'index']);

}

use : beforesave method ,

use : $registration->password=md5($registration->password);

flash data print

$this->Flash->success(\_\_('The registration has been saved.'));

$this->Flash->error(\_\_('The registration could not be saved. Please, try again.'));

Redirect

return $this->redirect(['controller'=>' ', 'action' => 'index']);

send data to view

$this->set('post', $post); // for single data send

$this->set(compact('registration')); // for multiple data send

$this->set('\_serialize', ['registration']);

---------------------------------------

another mehtod for multiple variable send

$this->set(compact('page', 'subpage', 'title\_for\_layout', 'data'));

/\* Display the list of data for the current page supplied by the controller \*/

echo '<ul>';

$data = $this->getVar('data');

foreach ($data['SomeData'] as $item) {

echo '<li>' . $item['text'] . '</li>';

}

echo '</ul>';

---------------------------

public function initialize()

    {

        parent::initialize();

        $this->loadComponent('Flash'); // Include the FlashComponent

    }

view or template

for link

<?= $this->Html->link('Add New Post', ['action' => 'add']) ?>

select all with condition

1) $registrations = $this->paginate($this->Registrations); // with pagination

or

2) $array = array(

'conditions' => ['firstname' => 'mit'],

'contain' => [],

'fields' => [],

'get' => 'all'

);

$this->Registrations->find($array['get'], $array)->all();

and send to view file

$this->set(compact('registrations'));

$this->set('\_serialize', ['registrations']);

on view file fetch by

<?php foreach ($registrations as $registration): ?>

<tr>

<!-- <td><?= $this->Number->format($registration->regid) ?></td> -->

<td><?= h($registration->firstname) ?></td>

<td><?= h($registration->lastname) ?></td>

<td><?= h($registration->email) ?></td>

<td><?= h($registration->Address) ?></td>

<td class="actions">

<?= $this->Html->link(\_\_('View'), ['action' => 'view', $registration->regid]) ?>

<?= $this->Html->link(\_\_('Edit'), ['action' => 'edit', $registration->regid]) ?>

<?= $this->Form->postLink(\_\_('Delete'), ['action' => 'delete', $registration->regid], ['confirm' => \_\_('Are you sure you want to delete # {0}?', $registration->regid)]) ?>

</td>

</tr>

<?php endforeach; ?>

3) if only one data or row get back then use first at place of all

$array = array(

'conditions' => ['regid' => $id],

'contain' => [],

'fields' => [],

'get' => 'all'

);

$registration=$this->Registrations->find($array['get'], $array)->first();

4) $user = $this->Users->getUser(array('email' => $this->request->data['email'], 'password' => md5($this->request->data['password'])));

selsect query when write in model with condition

$array = array(

'conditions' => ['email' => $param['email'],'password'=>$param['password']],

'contain' => [],

'fields' => [],

'get' => 'all'

);

$user=UsersTable::find($array['get'], $array)->first();

return $user;

or second rite

// $user = UsersTable::find()->where($param)->first();

// return $user;

5) $user = $this->Users->findById($this->request->data['id'])->first();

6) $this->Emails

->find('all')

->leftJoin('emails\_issues', 'emails\_issues.email\_id = Emails.id')

->where('emails\_issues.id IS NULL');

7)

Delete query

$this->request->allowMethod(['post', 'delete']); // allow method

$registration = $this->Registrations->get($id); // $id is id get by click on delete

if ($this->Registrations->delete($registration)) {

$this->Flash->success(\_\_('The registration has been deleted.'));

} else {

$this->Flash->error(\_\_('The registration could not be deleted. Please, try again.'));

}

return $this->redirect(['action' => 'index']);

update query

public function edit($id = null)

{

$registration = $this->Registrations->get($id, [

'contain' => []

]);

// echo $registration; exit();

if ($this->request->is(['patch', 'post', 'put'])) {

$registration = $this->Registrations->patchEntity($registration, $this->request->getData());

if ($this->Registrations->save($registration)) {

$this->Flash->success(\_\_('The registration has been saved.'));

return $this->redirect(['action' => 'index']);

}

$this->Flash->error(\_\_('The registration could not be saved. Please, try again.'));

}

$this->set(compact('registration'));

$this->set('\_serialize', ['registration']);

}

table sorting

<th scope="col"><?= $this->Paginator->sort('firstname') ?></th>

link with get method

<?= $this->Html->link(\_\_('View'), ['action' => 'view', $registration->regid]) ?>

link with post method

<?= $this->Form->postLink(\_\_('Delete'), ['action' => 'delete', $registration->regid], ['confirm' => \_\_('Are you sure you want to delete # {0}?', $registration->regid)]) ?>

id pass through post or get(link) then get their all data of particular id

$registration = $this->Registrations->get($id);

or

$registration = $this->Registrations->get($id, [

'contain' => []

]);

allow method

$this->request->allowMethod(['post', 'delete']);

for given link of external css & JS file

<?php

    echo $this->Html->charset();

    echo $this->Html->css('https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css');

    echo $this->Html->script('https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js');

?>

Make file of js in webroot---> js-->filename:styles.js then u can

Include one script file:

echo $this->Html->script('styles.js');

Include multiple script files:

echo $this->Html->script(['one.js', 'two.js']);

incase of forign key

$countries = $this->States->Countries->find('list', ['limit' => 200]);

ni jagyae muko aavu useful for dropdown

$countries = $this->States->Countries->find('list', ['keyField' => 'id', 'valueField' => 'countryname']);

$this->set(compact('state', 'countries'));

$this->set('\_serialize', ['state']);

form control; submition with curd operation,flash msg,debuging , singel/ multiple data send to view page,

external css,js link,link,postlink, paggination,sort table

LIST query with condition

$userRoles = $this->Users->find('list',['keyField' => 'id', 'valueField' => 'email','conditions'=>$this->request->data['conditions']]);

Deleteall

$this->MissingLexicons->deleteAll(['id IN' => $multiItemsArray]);

OR

$this->MissingLexicons->deleteAll(['id' => $multiItemsArray[]]);

MissingLexicons = Your Model name.

$multiItemsArray = Your deleted entities id

form helper

echo $form->create('User', array('action' => 'register'));

or to

<?= $this->Form->create($state) ?>

<fieldset>

<legend><?= \_\_('Add State') ?></legend>

<?php

//https://book.cakephp.org/2.0/en/core-libraries/helpers/form.html#FormHelper::hidden

echo $this->Form->control('countries\_id', ['options' => $countries]);

echo $this->Form->control('state\_name',array('class' => 'stay'));

echo $this->Form->textarea('notes',array('rows' => '5', 'cols' => '5'));

echo $this->Form->checkbox('done', array('value' => 555));

$options = array('M' => 'Male', 'F' => 'Female');

$attributes = array('legend' => false);

echo $this->Form->radio('gender', $options, $attributes);

echo $this->Form->create('User', array('type' => 'file'));

echo $this->Form->file('avatar');

echo $this->Form->button('A Button');

echo $this->Form->button('Another Button', array('type' => 'button'));

echo $this->Form->button('Reset the Form', array('type' => 'reset'));

echo $this->Form->month('mob');

// echo $this->Form->dateTime('dty', $dateFormat = 'DMY', $timeFormat = '12');

echo $this->Form->hidden('id');

?>

</fieldset>

<?= $this->Form->button(\_\_('Submit')) ?>

<?= $this->Form->end() ?>

or in view page

<h1>CakePHP Registration Form</h1>

<?php

echo $form->create('User', array('action' => 'register'));

echo $form->input('username');

echo $form->input('password', array('type' => 'password'));

echo $form->input('password\_confirm', array('type' => 'password'));

echo $form->submit();

echo $form->end();

?>

$.ajax({

dataType: "html",

type: "POST",

evalScripts: true,

url: '<?php echo Router::url(array('controller'=>'your-controller','action'=>'your-action'));?>',

data: ({type:'original'}),

success: function (data, textStatus){

$("#div1").html(data);

}

});

GET Working URL in CAKEPHP 3.x

'http://'.$\_SERVER['HTTP\_HOST'].$this->base

Mail in cake php 3.X

<http://findnerd.com/list/view/Sending-Mail-using-SMTP-in-Cakephp-3-0-/25576/>

https://book.cakephp.org/3.0/en/core-libraries/email.html#creating-reusable-emails

step:1

in config/app.php

replace this

'EmailTransport' => [

'default' => [

'className' => 'Mail',

// The following keys are used in SMTP transports

'host' => 'localhost',

'port' => 25,

'timeout' => 30,

'username' => 'user',

'password' => 'secret',

'client' => null,

'tls' => null,

'url' => env('EMAIL\_TRANSPORT\_DEFAULT\_URL', null),

],

by foolowing code

'EmailTransport' => [

'default' => [

'className' => 'Mail',

// The following keys are used in SMTP transports

'host' => 'localhost',

'port' => 25,

'timeout' => 30,

'username' => 'user',

'password' => 'secret',

'client' => null,

'tls' => null,

'url' => env('EMAIL\_TRANSPORT\_DEFAULT\_URL', null),

],

'gmail'=> [

'host' => 'ssl://smtp.gmail.com',

'port' => 465,

'username' => 'developer.technostacks@gmail.com', //your gmail address

'password' => 'Welcome12345', //your gmail password

'className' => 'Smtp',

'log' => true,

'context' => [

'ssl' => [

'verify\_peer' => false,

'verify\_peer\_name' => false,

'allow\_self\_signed' => true

]

]

],

],

step:2

in controller/ Appcontroller.php

call the library by in at top

use Cake\Network\Email\Email;

write the function below in aAppcontroller

public function sendUserEmail($to,$subject,$msg)

{

$email = new Email();

$email

->transport('gmail')

->from(['yourEmailID@domain.com' => 'yourEmailID@domain.com'])

->to($to)

->subject($subject)

->emailFormat('html')

->viewVars(array('msg' => $msg))

->send($msg);

return 1;

}

Step:3

in our controller where we have required to use email

public function sendmail()

{

$msg='Hellow User';

$subject='CakePHP3 Mail';

$to='rupesh@technostacks.com';

$true=$this->sendUserEmail($to,$subject,$msg);

if($true=='1')

{

$this->Flash->success(\_\_('Mail Send'));

return $this->redirect(['action' => 'index']);

}

else

{

$this->Flash->success(\_\_('Error'));

return $this->redirect(['action' => 'index']);

}

}

reusable mail in cake php 3.X

step:1

in config/app.php

replace this

'EmailTransport' => [

'default' => [

'className' => 'Mail',

// The following keys are used in SMTP transports

'host' => 'localhost',

'port' => 25,

'timeout' => 30,

'username' => 'user',

'password' => 'secret',

'client' => null,

'tls' => null,

'url' => env('EMAIL\_TRANSPORT\_DEFAULT\_URL', null),

],

by foolowing code

'EmailTransport' => [

'default' => [

'className' => 'Mail',

// The following keys are used in SMTP transports

'host' => 'localhost',

'port' => 25,

'timeout' => 30,

'username' => 'user',

'password' => 'secret',

'client' => null,

'tls' => null,

'url' => env('EMAIL\_TRANSPORT\_DEFAULT\_URL', null),

],

'gmail'=> [

'host' => 'ssl://smtp.gmail.com',

'port' => 465,

'username' => 'developer.technostacks@gmail.com', //your gmail address

'password' => 'Welcome12345', //your gmail password

'className' => 'Smtp',

'log' => true,

'context' => [

'ssl' => [

'verify\_peer' => false,

'verify\_peer\_name' => false,

'allow\_self\_signed' => true

]

]

],

],

Step 2 :

create the file src/Mailer/UserMailer.php.

Make folder Mailer & file UserMailer.php in this file write the below content

<?php

namespace App\Mailer;

use Cake\Mailer\Mailer;

class UserMailer extends Mailer

{

public function welcome($user)

{

// pr($user); exit();

$this

->setTo($user['email'])

->setSubject(sprintf('Welcome %s', $user['name']))

->setTemplate('default')

->setEmailFormat('html')

->setTransport('gmail')

->setViewVars(['firstname' => $user['name'],'useremailid' => $user['email']]);

// By default template with same name as method name is used.

// ->setLayout('custom');

}

public function resetPassword($user)

{

$this

->setTo($user->email)

->setSubject('Reset password')

->set(['token' => $user->token]);

}

public function register($user){

$this->setTo($user->email)

->setSubject('Welcome to Proximity Attendance' )

->setTemplate('register')

//->setLayout('default')

->setEmailFormat('html')

->setTransport('smtp')

->setViewVars(['firstname' => $user->firstname,'lastname' => $user->lastname,'verification\_code' => $user->verification\_code]);

}

}

Step 3 ;

in our controller where we have required to mail

3.1 add this

use Cake\Mailer\MailerAwareTrait;

3.2 write within class but not in any function below line

use MailerAwareTrait;

so looking like this

class UsersController extends AppController

{

use MailerAwareTrait;

3.3

here use in action of reusemail but u can use getMailer where u have required

note that in $this->getMailer('User')->send('welcome', [$user]);

public function reusemail()

{

$user = ['email' => 'rupesh@technostacks.com', 'name' => 'MR.BEan'];

$this->getMailer('User')->send('welcome', [$user]);

$this->Flash->success(\_\_('Mail Send'));

return $this->redirect(['action' => 'index']);

}

3.4 in src/template/email/html/default.ctp

<p>Hello</p>

<b>welcome to <?php echo $firstname; ?> </b></br>

<i>thanks & regards</i>

note: here $firstname declre in s>setViewVars(['firstname' => $user['name'],'useremailid' => $user['email']]) in usermailer file so u can use dynamics msg

LOGIN & LOGOUT USING AUTH IN CAKE

1st in app controller write in initialize function

$this->loadComponent('Auth', [

'authenticate' => [

'Basic' => [

'fields' => ['username' => 'email', 'password' => 'password'],

'userModel' => 'Users'

]

],

'loginAction' => [

'controller' => 'Users',

'action' => 'login'

]

]);

in our controller

public function login()

{

$params = $this->request->getData();

if ($this->request->is('post'))

{

if (isset($this->request->data['uesrname']) && !empty($this->request->data['uesrname']) &&

isset($this->request->data['password']) && !empty($this->request->data['password']))

{

$user = $this->Users->getUser(array('email' => $this->request->data['uesrname'], 'password' => md5($this->request->data['password'])));

if (empty($user) || $user == NULL)

{

$user['status'] = 0;

$user['msg'] = "Invalid username or password, try again.";

}

else

{

$user['status'] = 1;

$user['msg'] = "login successfully";

if ($user)

{

$this->Auth->setUser($user);

// $u=$this->Auth->user('password'); print\_r($u); exit(); to get autorised user data via this syntax scope only in controller

//echo $loguser=$this->request->Session()->read('Auth.User.firstname'); to get autorised user data via this syntex in view page only

$this->set(['users' => $user, '\_serialize' => ['users']]);

$this->redirect(['action' => 'index']);

}

}

}

}

}

public function logout()

{

$this->Flash->success('You are now logged out.');

return $this->redirect($this->Auth->logout());

}

in MODEL

public function getUser($param)

{

// First maethod

// $user = UsersTable::find()->where($param)->first();

// return $user;

// second method

$array = array(

'conditions' => ['email' => $param['email'],'password'=>$param['password']],

'contain' => [],

'fields' => [],

'get' => 'all'

);

$user=UsersTable::find($array['get'], $array)->first();

return $user;

}

SESSION IN CAKEPHP 3.X

$session = $this->request->session(); // strat the session

$session->write('Config.language', 'eng'); // set sesseion variable

<?php echo $this->request->session()->read('Config.language'); // read the session set value ?>

this->request->session()->delete('Some.value');

this->request->session()->check('Config.language');

this->request->session()->destroy();

COOKIES in CAKEPHP 3.X

in our controller action where we have to required

$this->Cookie->config('path', '/');

$this->Cookie->write('name', 'Larry');

echo $this->Cookie->read('name');

exit();

in APPcontroller

in action initialize()

write

$this->loadComponent('Cookie', ['httpOnly' => true]);

Note: cookies is use only in controller not in view bcoz it is a component not helper.

Only helper functionality exteds into view.

## Desktop notificcation using j query

## http://www.jqueryscript.net/other/jQuery-Plugin-For-HTML5-Desktop-Push-Notifications-easyNotify-js.html

Base url set

on view page

at top : use Cake\Routing\Router;

where to use

url:"<?php echo Router::url(array('controller'=>'States','action'=>'ajaxfileupload'));?>",

short way

url:"<?php echo $this->Url->build(['controller'=>'States','action'=>'ajaxfileupload']);?>",

AUTO COMPLATE

<http://www.smarttutorials.net/jquery-autocomplete-search-using-php-mysql-and-ajax/>

call function of model in controller

$email\_check = $this->Users->email\_check($this->request->data['email']);

here email\_check is function with argumrnt

**Theme Integration in CAKE PHP 3.X**

link: http://www.codexworld.com/cakephp-tutorial-part-3-working-with-elements-layout/

**Elements** are the small and reusable bits of view code.

**Layouts** are the template files that contain presentational code together with rendered Elements.

BaSIk IDEa get from link.

Step:1 boostrap html templae

step:2 create three files named head.ctp, header.ctp and footer.ctp.

head.ctp

<?php

    echo $this->Html->charset();

    echo $this->Html->css('https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css');

    echo $this->Html->script('https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js');

?>

<title>CakePHP Tutorial by CodexWorld</title>

header.ctp

header part put here

footer.ctp

footer part put here

step:3 put all 3 files at src/Template/Element/

step:4 Layouts are stored into the src/Template/Element/Layout/ directory. Create your custom layout src/Template/Element/Layout/frontend.ctp.

frontend.ctp. File content

<!DOCTYPE html>

<html lang="en">

<head>

    <?= $this->element('head') ?>

</head>

<body>

    <?= $this->element('header') ?>

*<!-- Page Content -->*

<div id="content" class="container">

        <?= $this->Flash->render() ?>

<div class="row">

            <?= $this->fetch('content') ?>

</div>

</div>

    <?= $this->element('footer') ?>

</body>

</html>

step:5 Use Custom Layout:

public function initialize(){

    parent::initialize();

    // Set the layout

    $this->viewBuilder()->layout('frontend');

}

in our controller , intialize fucnction ma write then set for only that controller & its method

when write in appcontroller's initialize function then set for all controller & all its methods

when u do not make frontend.ctp & make default.ctp file then no need to $this->viewBuider write at any where

u can crete another layout for login page & call it in usercontroller 's login method then layout is different for login & others.

Theme integrate with slick lab

step:1 src/template/layout

put 2 files default.ctp, login\_layout.ctp

step:2 src/template/element

put 4 files head.ctp,header.ctp,sidebar.ctp,footer.ctp

step:3 put/replce webroot folder

step:4 write in login method 's 1st line

$this->viewBuilder()->layout('login\_layout'); // Load Admin panel login look

<http://alvinalexander.com/php/cakephp-multiple-layouts-solution-controllers>

step:5 src/template/users

add or modified 2 files login.ctp and index.ctp using slick lab class of desinig.

Different login redirect

if($this->Auth->user('account\_type')=='admin'){

return $this->redirect(array('controller' => 'users', 'action' => 'admin\_index'));

}

elseif($this->Auth->user('account\_type')=='user'){

return $this->redirect(array('controller' => 'users', 'action' => 'view'));

}

else {

if ($this->Auth->login()) {

return $this->redirect($this->Auth->redirect());

}

$this->Session->setFlash(\_\_('Invalid username or password, try again'));

}

date

$monthday\_today = date('m-d');

$query = "SELECT \* FROM users WHERE YEAR(birthday) = YEAR(NOW()) AND MONTH(birthday) = MONTH(NOW())";

$query = "SELECT \* FROM users WHERE DATE\_FORMAT(birthday, '%m-%d') != :monthday\_today";

$query = "SELECT \* FROM users WHERE DATE\_FORMAT(birthday, '%m-%d') != ?";

SELECT \* FROM users WHERE DATE\_FORMAT(birthday, '%m-%d') != '04-2014'

$birthDate = '2017-04-06'; // Read this from the DB instead

$time = strtotime($birthDate);

if(date('m-d-y') == date('m-d-y', $time)) {

echo "They're the same!";

}

else{

echo "not same";

}

calculate age by date of birth

<?php

//An example date of birth.

$userDob = '1987-04-02';

//Create a DateTime object using the user's date of birth.

$dob = new DateTime($userDob);

//We need to compare the user's date of birth with today's date.

$now = new DateTime();

//Calculate the time difference between the two dates.

$difference = $now->diff($date);

//Get the difference in years, as we are looking for the user's age.

$age = $difference->y;

//Print it out.

echo $age;

Birth day between today & last date of current month

function getBirthdayDate()

{

$dateqry='';

$today=date("Y-m-d");

$lastdate=date("Y-m-t", strtotime($today));

$dateqry=" AND `dob`>= '".$today."' AND `dob`<='".$lastdate."'";

$sql="SELECT \* FROM `customers` WHERE store\_id='1389' ".$dateqry." ";

$result=dbQuery($sql);

return $row=$result->num\_rows;

}

NOW();

$today=date("Y-m-d");

$todaytime = date('H:i');

$two\_week = date('Y-m-d', strtotime($today.' +14 days'));

$lastdateofmonth\_of\_todaydate=date("Y-m-t", strtotime($today));

// Pear Mail Library

require\_once "Mail.php";

$from = '<fromaddress@gmail.com>';

$to = '<toaddress@yahoo.com>';

$subject = 'Hi!';

$body = "Hi,\n\nHow are you?";

$headers = array(

'From' => $from,

'To' => $to,

'Subject' => $subject

);

$smtp = Mail::factory('smtp', array(

'host' => 'ssl://smtp.gmail.com',

'port' => '465',

'auth' => true,

'username' => 'johndoe@gmail.com',

'password' => 'passwordxxx'

));

$mail = $smtp->send($to, $headers, $body);

if (PEAR::isError($mail)) {

echo('<p>' . $mail->getMessage() . '</p>');

} else {

echo('<p>Message successfully sent!</p>');

}

# Query Builder

Write in controller

use Cake\ORM\TableRegistry;

$articles = TableRegistry::get('Articles');

// Start a new query.

$query = $articles->find();

// Inside ArticlesController.php

$query = $this->Articles->find();

### Selecting Rows From A Table

use Cake\ORM\TableRegistry;

$query = TableRegistry::get('Articles')->find();

foreach ($query as $article) {

debug($article->title);

}

$query->where(['id' => 1]); // Return the same query object

$query->order(['title' => 'DESC']); // Still same object, no SQL executed

$query = $articles

->find()

->select(['id', 'name'])

->where(['id !=' => 1])

->order(['created' => 'DESC']);

foreach ($query as $article) {

debug($article->created);

}

object form

$resultsIteratorObject = $articles

->find()

->where(['id >' => 1])

->all();

foreach ($resultsIteratorObject as $article) {

debug($article->id);

}

array form

$resultsArray = $articles

->find()

->where(['id >' => 1])

->toArray();

foreach ($resultsArray as $article) {

debug($article->id);

}

### Selecting A Single Row From A Table

$article = $articles

->find()

->where(['id' => 1])

->first();

use Cake\ORM\TableRegistry;

$articles = TableRegistry::get('Articles');

$query = $articles ->find()

->select(['id', 'name','count' => $query->func()->count/sum/avg/min/max('\*')]) or ->select(['User.id', 'User.name'])

->select(['count' => $query->func()->count('\*')])

->distinct(['country'])

->innerJoinWith('Categories')

->leftJoin('emails\_issues', 'emails\_issues.email\_id = Emails.id')

->where(['id' => 1,'title' => 'First Post'])

->order(['created' => 'DESC']); or ->order(['title' => 'ASC', 'id' => 'ASC']);

->first();

->all(); or ->toArray();

->limit(50)

->page(2);

TreeBehavior

step 1: crete table in which i have required to add 3 coloumn parent\_id ,lft,rght and must be kept NULL.

step 2:put $this->addBehavior('Tree'); in model's initialize function

sytpe:3 in controller

$courses = $this->Courses;

$a = $courses->find('treeList', [

'keyPath' => 'url',

'valuePath' => 'name', // name of the field where in databse i mention name

'spacer' => '- '

]);

for whole & for particual parent id use below

$a = $courses->find('treeList', [

'keyPath' => 'url',

'valuePath' => 'name',

'spacer' => '- '

])->find('children', ['for' => 1]);

then in view file ...u can use in ul or in select to select as per your required ment

echo "<select>";

foreach ($a as $categoryName) {

echo "<option>".$categoryName . "\n</option>";

}

echo "</select>";

for breadcrubm

// $nodeId = 2;

// $crumbs = $courses->find('path', ['for' => $nodeId]);

// foreach ($crumbs as $crumb) {

// echo $crumb->name . ' > ';

// }

// pr($a);

FOR DROPDOWN SELECTED item AT TIME OF EDIT/UPDATE

in controller

$gift = $this->ApiHttpRequest->sendRequest(array("id" => $id));

$params = array(

'get' => 'list',

[

'keyField' => 'id',

'valueField' => 'title'

]

);

$nameof\_levels = $this->ApiHttpRequest->sendRequest($params,'getLevels');

$drop\_level\_name = $nameof\_levels;

$this->set('gift', $gift['giftcards']);

$this->set('drop\_level\_name', $drop\_level\_name );

$this->set('\_serialize', ['user']);

in view file

<div class="form-group ">

<label for="email" class="control-label col-lg-2">Levels \*</label>

<div class="col-lg-4">

<?= $this->Form->input('level\_id', ['options' => $drop\_level\_name,'label' => false,'default'=>$gift['level\_id']]); ?>

</div>

</div>

UPDATEALL 2 method

method 1

$conditions = array('id'=>$product\_id);

$fields = array('rating'=>$average\_rate);

ProductsTable::updateAll($fields, $conditions);

Method 2

use another table in another controller & update all

use Cake\Database\Expression\QueryExpression;

$users = TableRegistry::get('Users');

//pr($users); exit();

$expression = new QueryExpression('points = points + 1');

//pr($expression); exit();

$t1 =$users->updateAll([$expression],array('id' => $res['user\_id']));

for example

public function multiupdate()

{

if ($this->request->is(['patch', 'post', 'put']))

{

$data = $this->request->data;

// pr($data); exit();

// $data=array(16,17,18);

$expression = new QueryExpression('points = points + 1');

for($i=0;$i<count($data);$i++)

{

$t1 = $this->Users->updateAll([$expression],['id' => $data[$i]]);

//$t1 = $this->Users->updateAll([$expression],array('id' =>17,'id'=>16));

$datas=$data[$i];

}

//$t1 = $this->Users->updateAll([$expression],['id' => $data]);

$array = array(

'conditions' => ['id IN' =>$data],

'contain' => [],

'fields' => [],

'get' => 'all'

);

// pr($array); exit();

$users\_present=$this->Users->find($array['get'], $array)->all();

//pr($users\_present); exit();

$this->serialize(200,'points fields is updated',$users\_present );

//pr($t1);exit;

}

}

update multiple fields with condition (conditon ma id nhi pan biji fields hoy tyare) (UPDATE ON OTHER FIELDS Exclude id)

$requestData = $this->request->data;

//pr($requestData['status']); exit();

$conditions = $requestData['conditions'];

$fields = array('status'=>$requestData['status'],””=>””);

if ($this->Invitations->updateAll($fields, $conditions)) {

$this->serialize(200, 'Data is Edited');

}else{

$this->serialize(400, 'Data is not Edited');

}

## Saving Multiple Entities

Or update multiple entity of same row without condition

$data = [

[

'title' => 'First post',

'published' => 1

],

[

'title' => 'Second post',

'published' => 1

],

];

$articles = TableRegistry::get('Articles');

$entities = $articles->newEntities($data);

$result = $articles->saveMany($entities);

**Save query write in model**

$logsTable = \Cake\ORM\TableRegistry::get('Logs', array('table' => 'makes\_stores'));

$log = $logsTable->newEntity();

$log['make\_id']=$make\_ids[$i];

$log['store\_id']=$lastinset\_store\_id;

$logsTable->save($log);

Create CUSTOM HELPER IN CAKE PHP 3.X

Hey, It’s quite easy to create your own helper.You just need to follow some steps.

1. First go to the folder app/View/Helper and create a file name MytestHelper.php.  
2. Place the below code in the MytestHelper.php

<?php  
class MytestHelper extends AppHelper{  
public function testFunction($arg1){  
return ‘‘.$arg1.’‘;  
}  
}  
?>

3. Add this helper to controller where you want to use..  
Like:

class TestsController extends AppController{  
var $name = “Tests”;  
var $helpers = array(“Mytest”);  
…..  
……  
}

4. Now, you can use this helper in your view file.  
Like:

<?php  
echo ‘This helper is to make the content ‘.$this->Mytest->testFunction(‘bold’);  
///Will print “This helper is to make the content bold”.  
?>

for excel ,pdf,csv export

https://datatables.net/extensions/buttons/examples/initialisation/export.html

for coloumd visibility & export

https://datatables.net/extensions/buttons/examples/styling/bootstrap.html

for chartjs Helper(new task of cakephp)

https://github.com/netusco/CakePhp3-ChartJs/blob/master/src/View/Helper/ChartjsHelper.php

for foreign key belong to & has many

level id is a FK in gift table then

gift model

public function initialize(array $config)

{

parent::initialize($config);

$this->setTable('giftcards');

$this->setDisplayField('id');

$this->setPrimaryKey('id');

$this->addBehavior('Timestamp');

$this->belongsTo('Levels', [

'foreignKey' => 'level\_id',

'joinType' => 'INNER'

]);

}

level model

public function initialize(array $config)

{

parent::initialize($config);

$this->setTable('levels');

$this->setDisplayField('title');

$this->setPrimaryKey('id');

$this->addBehavior('Timestamp');

$this->hasMany('Giftcards', [

'foreignKey' => 'level\_id'

]);

}

link url build

<?php echo $this->Url->build(['controller'=>'Giftcards','action'=>'ajaxfileupload']);?>

link for css & for script

<?= $this->Html->css('bootstrap-fileinput-master/css/fileinput.css'); ?>

<?= $this->Html->script('bootstrap-fileinput-master/js/fileinput.js');?>

parameters

$params = array(

'conditions' => ['Users.id'=>$id],

'contain' => ['UserRoles'],

'fields' => array(),

'select' => array(),

'get' => 'all',

'order' => array(),

'page' => 1,

'limit' => 1

);

TREE for UL / category

https://stackoverflow.com/questions/8587341/recursive-function-to-generate-multidimensional-array-from-database-result/8587437#8587437

j query validation

http://code.runnable.com/UZJ24Io3XEw2AABU/how-to-validate-forms-in-jquery-for-validation

Quickblox

<https://quickblox.com/developers/Create_message_in_PHP>

https://quickblox.com/developers/Generate\_session\_in\_PHP

load model in controller in cakephp3

$this->loadModel('Media');

$this->Media->saveimage($product\_id,$pro\_image);

aggregate function in cakephp 3

$params = array(

'conditions' => ['product\_id'=>$product\_id,'rate != '=>''],

'contain' => [],

'fields' => array('rate'=>'avg(rate)'),

'select' => array()

);

$get\_avg=$this->Addevents->get\_avg($params);

$avg=AddeventsTable::find('all',$params)->first();

/\* ------------------------------------- --------------------- --------------------------------------- \*/

public function rate()

{

if ($this->request->is('post'))

{

$user\_id=$this->request->getData('user\_id');

$product\_id=$this->request->getData('product\_id');

$rate=$this->request->getData('rate');

$set\_param='rate';

//check user id,product id ,rate !=''

$is\_exist=$this->Addevents->check\_exist($user\_id,$product\_id,$rate,$set\_param);

//if exist in addevent tble then update in addevent tbl & (avg)rate update in product tbl

if($is\_exist['return']==true)

{

// update addevent tbl

$addevent\_id=$is\_exist['data']['0']->id;

$conditions = array('id'=>$addevent\_id);

$fields = array('rate'=>$rate);

$update\_rating=$this->Addevents->updateAll($fields, $conditions);

// get avg rate from addevent tbl

$params = array(

'conditions' => ['product\_id'=>$product\_id,'rate != '=>''],

'contain' => [],

'fields' => array('rate'=>'avg(rate)'),

'select' => array()

);

$get\_avg=$this->Addevents->get\_avg($params);

$average\_rate=$get\_avg->rate;

// update product tbl rate field

$this->loadModel('Products');

$this->Products->add\_rate($product\_id,$average\_rate);

$this->serialize(401, 'rating is updated.',$update\_rating);

}

else

{

//if not exist in addevent tble then add in addevent tbl & (avg)rate update in product tbl

$addevents = $this->Addevents->newEntity();

$addevents = $this->Addevents->patchEntity($addevents, $this->request->getData());

// add in addevent tbl

if ($this->Addevents->save($addevents))

{

// get avg rate from addevent tbl

$params = array(

'conditions' => ['product\_id'=>$product\_id,'rate != '=>''],

'contain' => [],

'fields' => array('rate'=>'avg(rate)'),

'select' => array()

);

$get\_avg=$this->Addevents->get\_avg($params);

$average\_rate=$get\_avg->rate;

// update product tbl rate field

$this->loadModel('Products');

$this->Products->add\_rate($product\_id,$average\_rate);

$this->serialize(200, 'The rate has been saved.',$addevents);

}else{

$this->serialize(500, 'Required all fields');

}

}

}

}

}

USE VENDOR/LIB IN COMPONENT

<http://findnerd.com/list/view/How-to-import-vendor-files-in-CakePHP-3-/18443/>

BEFORE FIND in model

use Cake\Utility\Text;

use Cake\Event\Event;

use ArrayObject;

public function beforeFind(Event $event, Query $query, ArrayObject $options, $primary){

$query->where(array('Collages.isDeleted' => 0));

}

public function beforeFind(Event $event, Query $query, ArrayObject $options, $primary){

$query->where(array('Collages.isDeleted' => 0));

}

PRIFIX FOR ROUNTING WHEN ADMIN folder make for cotroller of admin bt model ae comman

/\* ---------------- ROUTING FOR ADMIN PANEL--------------------- \*/

Router::prefix('admin', function (RouteBuilder $routes) {

$routes->connect('/', ['controller' => 'Users', 'action' => 'login']);

$routes->fallbacks(DashedRoute::class);

});

/\* ---------------- ROUTING FOR WEBSITE--------------------- \*/

Router::scope('/', function (RouteBuilder $routes) {

/\*\*

\* Here, we are connecting '/' (base path) to a controller called 'Pages',

\* its action called 'display', and we pass a param to select the view file

\* to use (in this case, src/Template/Pages/home.ctp)...

\*/

$routes->connect('/', ['controller' => 'Users', 'action' => 'login']);

|  |
| --- |
| [share](http://stackoverflow.com/a/34636677)[improve this answer](http://stackoverflow.com/posts/34636677/edit) |

<http://stackoverflow.com/questions/34150502/cakephp-3-different-login-redirection-depending-on-user-roles>

Multiple Inser Query

public function add()

{

// $c=Array(

// [

// "user\_id" => 1,

// "customer\_id" => 2,

// "item\_id" => 3,

// "notes" => "dfgrth"

// ],

// [

// "user\_id" => 1,

// "customer\_id" => 3,

// "item\_id" => 4,

// "notes" => "dfgrth"

// ]

// );

// echo "<pre>"; print\_r($this->request->data); exit();

if ($this->request->is('post'))

{

$entities = $this->Cart->newEntities($this->request->data);

$result = $this->Cart->saveMany($entities);

if (count($result)>0) {

$this->serialize(200, ' Record saved',$result);

}else{

$this->serialize(500, 'Required all fields');

}

}

}

Simple Query use in cakephp 3.X

put on TOP

use Cake\Datasource\ConnectionManager;

In Function -----------------

$conn = ConnectionManager::get('default'); $sql = " Select s.\*,u.firstname,u.lastname from suggestion as s LEFT JOIN users as u ON s.user\_id = u.id where s.id=" . $id . " and s.isDeleted=0 "; $stmt = $conn->execute($sql); $results = $stmt->fetchAll('assoc');