**1000**

**1002 (Home page start search form)**

After the user submits the form in the home page, the form datas are submitted to the url : <http://dinendash.ca/search> with get method with names ‘city’ for city, ‘cusine ’ for cuisine and ‘name’ for Restaurant name. This refers to the index function of search controller which runs following codes in order to retrieve the datas:

The same function is called even for module **1100,** from where we get additional field of maximum party size with name ‘max\_party\_size’ and Reservation time with name ‘time’

if(isset($\_GET['name'])) // since name is a required field, we check if the form is submitted using the name field

{

// Retrieving of search fields

$name = trim($\_GET['name']);

$location = $\_GET['city'];

$cuisine = $\_GET['cuisine'];

$size = isset($\_GET['max\_party\_size']);

$time = $\_GET['time'];

if($name)

{

$arr['LOWER(name) LIKE'] = '%'.trim(strtolower($name)).'%'; /\* user might search regardless of alphabet cases so in order to match the cases we change botht the request and database value to lowercase with this code\*/

}

if($location)

{

$arr['LOWER(city) LIKE'] = '%'.strtolower($location).'%';

}

if($cuisine)

{

$arr['LOWER(cuisine) LIKE'] = '%'.strtolower($cuisine).'%';

}

if($size)

{

if(!is\_numeric($size))

$size = 0;

$arr['LOWER(max\_party\_size) >='] = trim(strtolower($size));

}

$arr['active'] = 1;

$page = $\_GET['page'];

// For the addition fields in search page

if(isset($\_GET['search']) && $time!="Time")

{

$time=date('H:i:s',strtotime($time.':00'));

$time=strtotime($time);

$data=array();

$data=$this->Restaurant->find('all',array('conditions'=>$arr));

$search=array();

foreach($data as $d)

{

$ti[]=$d['Restaurant']['reservation\_start'];

}

$count=sizeof($ti);

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

{

$tim=strtotime($ti[$i]);

$ti[$i]=$tim-$time;

$ti[$i]=abs($ti[$i]);

}

$da=$ti;

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

{

for($j=$i+1;$j<$count;$j++)

{

if($da[$i]>$da[$j])

{

$temp=$da[$i];

$da[$i]=$da[$j];

$da[$j]=$temp;

}

}

}

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

{

for($j=0;$j<$count;$j++)

{

if($da[$i]==$ti[$j])

{

$search[$i]=$data[$j];

}

}

}

$this->set('search',$search);

$this->set('restaurantss',$search);

}

else

{

$this->set('search',$this->Restaurant->find('all',array('conditions'=>$arr,'limit'=>$page.',15')));

//Here restaurantss is set as variable in the view file for the fetched restaurant meeting the user’s criteria

$this->set('restaurantss',$this->Restaurant->find('all',array('conditions'=>$arr,'limit'=>$page.',15')));

}

$this->set('count',$this->Restaurant->find('count',array('conditions'=>$arr)));// In order to display the number of restaurants found

}

**1006 (Home page slider)**

**To retrieve the slider images of the home page we have requested a url from the view file index of restaurant folder which is :**

$slider = $this->requestAction($req.'/restaurants/getslider');

This calls the function get slider inside restaurants controller which performs following execution:

$this->loadModel('Imageslide');

return $this->Imageslide->find('all');

This two lines fetches all the slider images uploaded by admin which are displayed in the home page (i.e index view of restaurants controller) using these codes

if($slider)

foreach($slider as $s)

{

$sl++;

?>

<div class="item<?php if($sl==1){?> active<?php }?>">

<img src="<?php echo $req;?>/img/slider/<?php echo $s['Imageslide']['file'];?>" alt="" />

</div>

<?php

}

?>

**1004 (quick links in the footer)**

Quick links are manually placed in the footer of the website without any logical operation using these static codes

<div class="footleft" >

<span class="title">Quick Links</span> <br />

<a href="<?php echo $req;?>">Home</a> &nbsp; <br />

<a href="<?php echo $req;?>/restaurants/reservation\_key">Reservation</a> &nbsp; <br />

<a href="<?php echo $req;?>/search">Search</a> &nbsp; <br />

<a href="<?php echo $req;?>/pages/about">About</a> &nbsp; <br />

<a href="<?php echo $req;?>/pages/help">Help / FAQ</a> &nbsp; <br />

<a href="<?php echo $req;?>/pages/terms">Terms of Use</a> &nbsp; <br />

<a href="<?php echo $req;?>/pages/privacy">Privacy Policy</a> &nbsp; <br />

<div class="clear"></div>

</div>

<div class="footleft">

<span class="title">Restaurants</span><br/>

<?

if($this->Session->read('res\_id'))

{

?>

<a href="<?php echo $req;?>/reservations">Dashboard</a> &nbsp; <br />

<a href="<?php echo $req;?>/restaurants/logout">Logout</a> &nbsp; <br />

<?}else{?>

<a href="<?php echo $req;?>/restaurants/register">Register</a> &nbsp; <br />

<a href="<?php echo $req;?>/restaurants/login">Login</a> &nbsp; <br />

<?}?>

</div>

<div class="footleft" >

<span class="title">Contact Us</span> <br />

<a href="<?php echo $req; ?>/pages/contact">Contact Us</a> &nbsp; <br />

<a href="<?php echo $req; ?>/pages/report\_bug?p=<?echo $\_SERVER['HTTP\_HOST'] . $\_SERVER['REQUEST\_URI']?>">Report a Bug</a> &nbsp; <br />

<a href="<?php echo $req; ?>/restaurants/suggest">Suggest a Restaurant</a> &nbsp; <br />

<div class="clear"></div>

</div>

**1010 (Reservation check form [shifted to header in the new design])**

When the user enters the booking id and hits view button, the booking id is submitted to the check function of the reservation controller, which makes following execution:

//To retrieve the booking id

if(isset($\_POST['key']))

{

$k = trim($\_POST['key']);

}

//To check the validity of the booking id

if($r = $this->Reskey->find('first',array('conditions'=>array('reser\_key'=>$k)))) //enter into this conditions only if the booking id is found

{

// To check if the reservation is already cancelled

$ca = $this->Reservation->findById($r['Reskey']['reser\_id']);

if($ca['Reservation']['cancelled']=='2')

{

$this->Session->setFlash("The host has already cancelled the reservation.");

$this->redirect('/restaurants/reservation\_key/0');

}

else //if the reservation is not cancelled

{

$this->set('resk',$r);

$this->set('key',$k);

}

}

else

{

// if the booking id is not found

$this->redirect('/restaurants/reservation\_key/0/res\_error');

}

**1100**

**Already described in module 1000**

**1200 (Restaurant detail page)**

**When we view any restaurant, it fetches the information from the detail function of Restaurants controller:**

**1202-1204 (Restaurant info)**

**The associated restaurant information is fetched from this code:**

$res = $this->Restaurant->find('all',array('conditions'=>array('slug'=>$slug)));

$res\_id = $res[0]['Restaurant']['id'];

**The restaurant image is fetched using this code:**

$this->set('imgs',$this->Image->find('all',array('conditions'=>array('res\_id'=>$res\_id))));

**1206**

**The associated featured menu items and categories are fetched using this code:**

$cat = $this->Menu\_category->find('all',array('conditions'=>array('res\_id'=>$res\_id)));

**The menu categories are then used in the same function to fetch the menu items using this code.**

**foreach($cat as $c)**

**{**

**$m = $this->Menu->find('all',array('conditions'=>array('res\_id'=>$res\_id,'cat\_id'=>$c['Menu\_category']['id'],'featured'=>'0'),'order'=>'display\_order ASC'));**

**if($m)**

**{**

**$ca['cat\_id'][] = $c['Menu\_category']['id'];**

**$ca['cat\_title'][] = $c['Menu\_category']['title'];**

**$count = count($ca['cat\_id'])-1;**

**foreach($m as $me)**

**{**

**$menus[$count]['id'][] = $me['Menu']['id'];**

**$menus[$count]['price'][] = $me['Menu']['price'];**

**$menus[$count]['menu\_item'][] = $me['Menu']['menu\_item'];**

**$menus[$count]['description'][] = $me['Menu']['description'];**

**$menus[$count]['active'][] = $me['Menu']['active'];**

**}**

**}**

**}**

**if(isset($menus))**

**{**

**$this->set('cat',$ca);**

**$this->set('menu',$menus);**

**}**

**The pdf menu are fetched using this code:**

$this->set('pdfs',$this->MenuPdf->find('all',array('conditions'=>array('res\_id'=>$res\_id))));

**1208 (shifted to home page)**

**This images are not dynamically fetched rather placed manually in the view file as this:**

<div class="arrowHolder2 clearfix">

<table cellspacing="0" cellpadding="0" border="0" class=""><tr>

<td class="blue WhiteCircle">1</td>

<td class="blue">FIND A RESTAURANT</td>

<td class="BlueGrayArrow"></td>

<td class="GrayCircle">2</td>

<td class="">PLACE YOUR ORDER</td>

<td class="GrayGrayArrow"></td>

<td class="GrayCircle">3</td>

<td class="">INVITE YOUR FRIENDS</td>

</tr></table>

</div>

**1300 (Restaurant reservation page)**

**This page calls the reservation function of restaurants controller.**

**1302-1304**

**The restaurant info are fetched into this page with following codes:**

$res = $this->Restaurant->find('first',array('conditions'=>array('slug'=>$slug))); // Here slug is obtained from the url

$this->set('res',$res);

**1306**

**After the user places desired value for his reservation and hits the button to submit, followin execution is made in the same function:**

if(isset($\_POST['submit'])) // This is to test if the user has submitted the form

{

$arr['res\_id'] = $res['Restaurant']['id'];

$arr['reservation\_date'] = date('Y-m-d',strtotime($\_POST['reservation\_date']));

$arr['reservation\_time'] = date('H:i:s',strtotime($\_POST['time'].':00'));

$arr['party\_size'] = $\_POST['party\_size'];

$arr['fullname'] = $\_POST['full\_name'];

$arr['email'] = $\_POST['email'];

$this->Session->write('reser\_email',$arr['email']);

$arr['phone'] = $\_POST['phone\_number'];

if(isset($reser\_id))

{

$this->Reservation->id=$reser\_id;

foreach($arr as $k=>$v)

{

$this->Reservation->saveField($k,$v); // this is to save the reservation made by the user if the user is editing the already made reservation

}

}

else

{

$this->Reservation->create();

$this->Reservation->save($arr); // This is to make a brand new reservation

$this->Session->write('reser\_id',$this->Reservation->id);

}

// To make necessary redirects after the reservation is made

if(isset($\_POST['reserve\_id']) && $\_POST['reserve\_id']!='')

{

$this->redirect('reservation\_edit/'.$slug.'/'.$\_POST['reserve\_id']);

}

else

{

$this->redirect('reservation\_success/'.$slug.'/'.$this->Reservation->id);

}

}

**1400 (Food ordering page)**

**This page calls a function called pre\_order of Restaurants controller where all the menu items and categories are fetched and shown in the page so that the user can choose the quantity and make the order**

**1404-1412**

**This code is to find out the related restaurant information:**

$res = $this->Restaurant->find('first',array('conditions'=>array('slug'=>$slug)));

**This is the code to determine whether the menu belongs to dinner, lunch or both and to get all the menu categories of the menu:**

if(isset($\_GET['both']))

{

$category = $this->Menu\_category->find('all',array('conditions'=>array('res\_id'=>$res['Restaurant']['id']),'order'=>'`order`'));

}

elseif(isset($\_GET['lunch']))

{

$category = $this->Menu\_category->find('all',array('conditions'=>array('res\_id'=>$res['Restaurant']['id'],'OR'=>array(array('cattype'=>1),array('cattype'=>0))),'order'=>'`order`'));

$this->set('lun','1');

}

elseif(isset($\_GET['dinner']))

{

$category = $this->Menu\_category->find('all',array('conditions'=>array('res\_id'=>$res['Restaurant']['id'],'OR'=>array(array('cattype'=>2),array('cattype'=>0))),'order'=>'`order`'));

$this->set('din','1');

}

else

{

$category = $this->Menu\_category->find('all',array('conditions'=>array('res\_id'=>$res['Restaurant']['id'],'OR'=>array(array('cattype'=>1),array('cattype'=>0))),'order'=>'`order`'));

}

$this->set('category',$category);

**These are the codes to determine the extra items and menu items displayed:**

$this->set('ext',$this->Extra);

$this->set('menu',$this->Menu->find ('all',array('conditions'=>array('res\_id'=>$res['Restaurant']['id']),'order'=>'display\_order')));

// After getting the categories, menu item and extra item other necessary logics are implemented in the view in order to manage them in proper order

**This is the code to get the reservation detail if the user logged in has already made the reservation in the same restaurant**

$this->set('reser',$this->Reservation->find('first',array('conditions'=>array('id'=>$id))));

**1500 (Credit card detail page no more exists, rather we have a thankyou page with send invitation option)**

**In this page user who made reservation can send out invitation to the user with booking id: An ajax request (request without page refresh) is made to the send\_book function of restaurants controller where following executions are made**

$em = $\_POST['em']; //In order to fetch the comma seperated email values

$id = $\_POST['reser\_id']; // In order to get the reservation id

$rand = $\_POST['rand']; // In order to get the booking id

$q\_reservation = $this->Reservation->findById($id); // using reservation id the reservation detail is fetched using this code

$res\_id = $q\_reservation['Reservation']['res\_id']; // to get the restaurant id from the associated reservation id

$this->Restaurant->id = $res\_id; // To fetch the restaurants detail

$current\_restro = $this->Restaurant->read();

// These are the url to be sent out in the email so that user can click on it to go to the required page

$url2 = 'http://www.dinendash.ca/reservations/check/'.$rand;

$url1 = 'http://www.dinendash.ca';

$this->loadModel('Reservation');

$arr = explode(',',$em); // this is to separate comma seperated emails to individual emails in an array

$self = $\_POST['self']; // this is the self email

if($self)

{

$arr[] = $self;

}

$msg = $\_POST['msg'];

$title = $\_POST['title'];

$ok = 0;

// This is to loop through each comma separated emails to send each one a copy of invitation

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

{

if(trim($arr[$i])!= "")

{

if($\_SERVER['SERVER\_NAME']=='localhost')

{

$url = 'http://localhost/dinendash/img/';

}

else

{

$url = 'http://dinendash.ca/img/';

}

$q = $this->Reservation->find('first',array('conditions'=>array('id'=>$id)));

$fullname = $q['Reservation']['fullname'];

$from = (array('info@dinendash.ca' => 'Dine N Dash'));

$to = (trim($arr[$i]));

$sub = ("Dine N' Dash - You're invited to an event!");

$msg = ("<div style=' overflow: hidden; background: #fff; margin: 0 auto; max-width: 700px; padding: 10px; text-align: center; -webkit-border-radius: 5px; -moz-border-radius: 5px; border-radius: 5px; -webkit-box-shadow: rgba(0,0,0,0.2) 0px 1px 3px; -moz-box-shadow: rgba(0,0,0,0.2) 0px 1px 3px; box-shadow: rgba(0,0,0,0.2) 0px 1px 3px;font-family: Georgia, serif;font-size:18px;border:2px solid #c1c1c1;color:#222;'>

<img src='".$url."g1.jpg' style='width:80px; float:left;' />

<img src='".$url."g2.jpg' style='width:80px; float:right;' />

<br>

<center>

<img src='http://dinendash.ca/img/logo\_color.png' />

</center>

<br><br>

Who?<br>

<b>Hosted by ".$q\_reservation['Reservation']['fullname']."

</b><br><br>

Where?

<br>

<b>

"

.$current\_restro['Restaurant']['name'].", "

.$current\_restro['Restaurant']['street'].", "

.$current\_restro['Restaurant']['city'] ." "

.$current\_restro['Restaurant']['prov\_state']." "

.$current\_restro['Restaurant']['pos\_zip']

."

</b>

<br><br>

When?

<br>

<b>

"

.$q\_reservation['Reservation']['reservation\_date']." at "

.date ('g:i A',strtotime($q\_reservation['Reservation']['reservation\_time']))

."

</b><br><br>

~

<br>

Log into <a href='".$url1."'>www.dinendash.ca</a> with booking ID: <b><u>".$rand."</u></b>

<br>

or

<br>

Click <a href='".$url2."'>here</a> to make your order now.<br>

~

<br><br>

Special Message from the Host:

<br>

<span style='padding:0 20px;font-weight:bold;'>".stripslashes(nl2br($msg))."

</span>

<br><br>

-The Dine N' Dash Team

<br>

<img src='".$url."g3.jpg' style='width:80px; float:left;' />

<img src='".$url."g4.jpg' style='width:80px; float:right;' />

<br>

</div>

");

AppController::Send\_Email($from,$to,$sub,$msg);

unset($from,$to,$sub,$msg);

$ok = 1;

}

}

$this->Session->setFlash("Invitation(s) sent successfully. Thank you for using Dine N' Dash!");

echo $ok;

die();

**1600**

**1602-1608 (Same as 1400)**

**1700 (Restaurant registration page)**

**As the user fills up the form and hits enter following codes run in restaurants register function:**

**//Retrieving the user submitted values for restaurant like name, address, cuisine, website, images etc**

$arr['name'] = $\_POST['name'];

$arr['street'] = $\_POST['street'];

$arr['prov\_state'] = $\_POST['prov'];

$arr['city'] = $\_POST['city'];

$arr['cuisine'] = $\_POST['cuisine'];

$arr['website'] = $\_POST['website'];

$arr['pos\_zip'] = $\_POST['pos\_zip'];

$arr['latitude'] = $\_POST['latitude'];

$arr['longitude'] = $\_POST['longitude'];

$arr['reservation\_start'] = $\_POST['reservation\_start'];

$arr['reservation\_end'] = $\_POST['reservation\_end'];

$arr['reservation\_start'] = date('H:i:s',strtotime($arr['reservation\_start'].':00'));

$arr['reservation\_end'] = date('H:i:s',strtotime($arr['reservation\_end'].':00'));

$arr['HST']=$\_POST['hst'];

$arr['phone'] = $\_POST['phone'];

**// This is to create a unique slug for the restaurnat**

$pattern = '/[^a-zA-Z0-9-' . $whiteSpace . ']/u';

$cleared = preg\_replace($pattern, '', (string) $arr['name']);

$conditions = $this->postConditions(array('slug'=>$cleared));

$ch = $this->Restaurant->find('all',array('conditions'=>array('slug'=>$cleared)));

if(empty($ch))

{

$arr['slug'] = $cleared;

}

$arr['picture'] = $\_POST['picture'][0]; **// This is to get the main image of the restaurant**

$this->Restaurant->create();

$this->Restaurant->save($arr); **// This is to save the Restaurant to the database**

$arrs['res\_id'] = $this->Restaurant->id;

if(!isset($arr['slug']))

{

$id = $this->Restaurant->id;

$this->Restaurant->read();

$arr['slug'] = $cleared.'-'.$id;

$this->Restaurant->save(array('slug'=>$arr['slug'])); **// This is to save the associated slug to the restaurant**

$arrs['res\_id'] = $this->Restaurant->id;

}

**// This is to save all other images to the database for the restaurant**

if(count($\_POST['picture'])>0)

{

$this->loadModel('Image');

for($i=0;$i<=(count($\_POST['picture'])-1);$i++)

{

$arrs['images'] = $\_POST['picture'][$i];

if($arrs['images'])

{

$this->Image->create();

$this->Image->save($arrs);

}

}

}

**1800 (Menu Category and menu creation)**

**The left side bar buttons in the dashboard after the restaurant login are just a static buttons that links to its related pages**

**1802**

<li class=" btn-inverse <?php if($act == 'reserve'){?>active<?php }?>"><a href="<?php echo $req;?>/reservations">Reservation Manager</a></li>

**1804**

<li class=" btn-inverse <?php if($act == 'menu'){?>active<?php }?>"><a href="<?php echo $req;?>/restaurants/manage\_menu">Menu Manager</a></li>

**1806**

<li class=" btn-inverse <?php if($act == 'setting'){?>active<?php }?>"><a href="<?php echo $req;?>/restaurants/setting">Account Settings</a></li>

**1808**

<li class=" btn-inverse <?php if($act == 'notification'){?>active<?php }?>"><a href="<?php echo $req;?>/restaurants/notifications">Notifications</a></li>

**1810 (Creating menu category)**

**Once the user provides the name of a category and hits create button: the add\_menu\_cat function of restaurants controller is called where following execution is made:**

**// Slug is obtained from the url which helps us to find out the restaurant associated to the menu**

if($slug)

{

$res = $this->Restaurant->find('first',array('conditions'=>array('slug'=>$slug)));

$res\_id = $res['Restaurant']['id']; **// obtaining restaurant id using slug**

}

else

{

$res\_id = $this->Session->read('res\_id'); **// if the slug is not found we use the id saved in the** session to identify the restaurant

}

$cat['res\_id'] = $res\_id; **// assigning the restaurant id to the category table of the database**

$cat['title'] = $\_POST['cat']; **// retrieving the category name inserted by the user**

$cat['description'] = $\_POST['description']; **// Retreiving the category description**

**// This is to assign the display order to the newly created menu category**

$q = $this->Menu\_category->find('first', array('conditions'=>array('res\_id'=>$res\_id),'order'=>'`order` DESC'));

if($q)

{

$order = $q['Menu\_category']['order'];

}

else

{

$order = 1;

}

$order++;

$cat['order']=$order;

**// Saving the newly created menu to the database**

$this->Menu\_category->create();

$this->Menu\_category->save($cat);

$this->Session->setFlash('Category successfully created!');

**1812 (Menu creation)**

**Once user fill up the form to create the menu and hits enter, the addmenu() function of restaurants controller is called. Following execution takes place here:**

**// Retrieving the menu values from the form**

$arr['cat\_id'] = $\_POST['data']['Menu']['cat\_id']; **// this is to get the associated category for the menu chosen by the user**

$arr['res\_id'] = $id; **// this is to get the associated restaurant of the menu which is retrieved from the url**

$arr['menu\_item'] = $\_POST['data']['Menu']['menu\_item']; **//This is to retrieve the menu item name**

$arr['price'] = $\_POST['data']['Menu']['price']; **//This is to retrieve the menu item price**

**// Ordering the menu by assigning the display order**

$ord = $this->Menu->find('first',array('conditions'=>array('cat\_id'=>$arr['cat\_id'],'res\_id'=>$arr['res\_id']),'order'=>'display\_order DESC'));

if($ord)

{

$arr['display\_order'] = $ord['Menu']['display\_order']+1;

}

else

{

$arr['display\_order'] = 1;

}

**// Saving the menu to the database**

if($this->Menu->save($arr))

{

if(isset($\_POST['data']['Menu']['has\_complimentary']))

{

$this->Menu->saveField('has\_complimentary','1');

}

$arr['menu\_id'] = $this->Menu->id;

**// Saving the extra field associated to the menu**

if(isset($\_POST['title']))

{

$this->Menu->saveField('has\_regular',1);

for($i=0;$i<=(count($\_POST['title'])-1);$i++)

{

$arr['title'] = $\_POST['title'][$i];

$arr['price'] = $\_POST['price'][$i];

$this->Extra->create();

$this->Extra->save($arr);

}

}

}

**1900 (Reservation manager module)**

**The index function of reservations controller is called in order to get the reservation list of the logged in restaurant admmin**

**1902-1918**

**// The reservation could be listed with or without date filter**

**// If particular date is selected**

if($date)

{

$arr = explode('\_',$date);

$this->set('y',$arr[0]); **// getting the year value from the date**

$this->set('m',$arr[1]); **// getting the month value from the date**

}

**// To check if non admin is trying to browse the page and to restrict non-admin from this page**

$this->check\_session();

$res\_id = $this->Session->read('res\_id');

**// if date is not selected the date is set to be today’s date**

if(!$date)

{

$dates = date('Y-m-d');

$cond = array('res\_id'=>$res\_id,'reservation\_date LIKE '=>$dates."%",'parent\_of'=>0,'id IN (SELECT reser\_id FROM reskeys)','cancelled <>'=>'1'); **/\* Preparing the condition to get the reservations in accordance to restaurant id, selected date and valid reservation. The validity of reservation is supposed only if booking id/ Reservation key is created for that particular reservation \*/**

}

else

{

**// Preparing above condition for the user defined date**

$cond = array('res\_id'=>$res\_id,'reservation\_date'=>$date,'parent\_of'=>0,'id IN (SELECT reser\_id FROM reskeys)','cancelled <>'=>'1');

}

**// getting all the reservation using above conditions and sending them to reservation listing view in a paginated format**

$this->paginate = array('conditions'=>$cond,'order'=>'reservation\_date DESC','limit'=>'10');

$this->set('item',$this->paginate('Reservation'));

**2000 (Reservation detail page)**

**This modules uses the detail() function of reservations controller**

**2002-2008**

**Getting user and reservation detail**

**The user detail in this page is retrieved by fetching these information from the reservation table using this code:**

$ord = $this->Reservation->find( 'all',array('conditions'=>array('OR'=>array('id'=>$id,'parent\_of'=>$id)),'order'=>'parent\_of,reservation\_date ASC'));

**//according to this code the parent reservation, i.e the reservation made for the first time and all other reservation made using same booking id by the the invited users are alll merged.**

**2010**

**// In the same fetched datas (from reservation table), we also have comma seperated menu item id, by which menus are fetched using these codes in the view file, and using those menu ids item name, price etc are fetched in the detail view file of reservations module:**

<?php $mf = $menu->find('first',array('conditions'=>array('id'=>$m)));echo $mf['Menu']['menu\_item']?>

**2012**

**// In order to approve the reservation, when the user clicks on Confirm button on the same page, approve() function of reservations controller is called, which perfoms following execution**

$this->Reservation->id = $id; **// associating the reservation row with the help of id from url**

$this->Reservation->saveField('approved',1); **// saving the approved field to 1 (which means approved)**

$this->Reservation->saveField('cancelled',0); **// saving the cancelled field to 0 (which means cancelling the cancelled state if any)**

**//Finding all the child reservation associated to the main reservation, and setting them and approved and non cancelled**

$other = $this->Reservation->find('all',array('conditions'=>array('parent\_of'=>$id)));

foreach($other as $o)

{

$this->Reservation->id = $o['Reservation']['id'];

$this->Reservation->saveField('approved',1);

$this->Reservation->saveField('cancelled',0);

}

**2100 (Van, please do it)**