



First, you need to have the following pre-requisites: a web server of your choice (IIS or Apache), PHP >= version 5.0, CI 1.5.4 installed, Xajax 0.5.3 library installed in CI, Nusoap 1.94 installed in CI, jquery 1.1.4 library, a Mysql server >= version 5.0 (the example requires a mysql server with stored procedures support), Firefox with Firebug plugin (for debugging).

Step 1. Create a table named "members" with the following table schema:

```
-- Table "members" DDL

CREATE TABLE `members` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `firstname` varchar(255) CHARACTER SET latin1 NOT NULL,
  `lastname` varchar(255) CHARACTER SET latin1 NOT NULL,
  `birthday` date NOT NULL DEFAULT '0000-00-00',
  PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=12 DEFAULT CHARSET=utf8 CHECKSUM=1
```

Fill the table with some data:

id	firstname	lastname	birthday
1	John	Smith	1978-05-03
2	John	Doe	1980-10-10
3	Jane	Doe	1975-02-07
4	Juan	Perez	1950-09-30
5	Jean	Dupont	1990-08-24
6	Mario	Rossi	1943-12-04
7	Jan	Novak	1963-11-16
8	Joe	Borg	1986-05-01
9	Paul	Martin	1965-04-13
10	Jaan	Tamm	1971-05-19
11	Ion	Popescu	1976-01-21

Step 2. Create 3 stored procedures that will be required for data selection:

```
CREATE PROCEDURE `MYSQL_select_members`(IN _start INT, IN _limit INT,
IN _sortField VARCHAR(20), IN _sortOrder VARCHAR(4))
BEGIN
  DECLARE _SQLOrder VARCHAR(255);
```



```
DECLARE _SQLLimit VARCHAR(255);

IF (_start >= 0 AND _limit > 0) THEN
    SET _SQLLimit = CONCAT(" LIMIT ", _start, ", ", _limit);
ELSE
    SET _SQLLimit = "";
END IF;

IF (_sortField <> '' AND _sortOrder <> '') THEN
    SET _SQLOrder = CONCAT(" ORDER BY ", _sortField, " ",
_sortOrder);
ELSE
    SET _SQLOrder = "";
END IF;

SET @SQL = CONCAT("SELECT * FROM `members` ", _SQLOrder,
_SQLLimit);
PREPARE _SQL FROM @SQL;
EXECUTE _SQL;
DEALLOCATE PREPARE _SQL;
END
```

```
CREATE PROCEDURE `MYSQL_select_member_info`(IN _id INT)
BEGIN
    SELECT * FROM `members` WHERE id = _id;
END
```

```
CREATE PROCEDURE `MYSQL_count_members`()
BEGIN
    SELECT COUNT(*) AS `count` FROM `members`;
END
```

Step 3. Create a model named "member.php":

```
<?php
class Member extends Model{
    var $_start = null;
    var $_limit = null;
    var $_sortField = null;
    var $_sortOrder = null;
```



```
function __construct(){
    parent::Model();
}

// note the "MODEL_" prefix in order to distinguish this
select_members from the other functions named alike that may occur

function MODEL_select_members() {
    $members = array(); // initialize "members" variable

    $this->_start = isset($this->_start)?$this->_start:0;
    $this->_limit = isset($this->_limit)?$this->_limit:5;
    $this->_sortField = isset($this->_sortField)?$this->
    >_sortField:"lastname";
    $this->_sortOrder = isset($this->_sortOrder)?$this->
    >_sortOrder:"ASC";

    if($this->_connect() === true){
        // mysql server 5.0 is a common thing nowadays so i prefer
stored procedures instead of dynamic sql
        // note the "MYSQL_" prefix in order to distinguish this
select_members from the other functions named alike that may occur

        $sql = "CALL MYSQL_select_members(".$this->_start.",
        ".$this->_limit.", '".$this->_sortField."', '".$this->_sortOrder."')";
        $query = $this->db->query($sql);

        if($query->num_rows() > 0){
            foreach($query->result_array() as $row){
                array_push($members, $row);
            }
        }

        $query->free_result();

        // in case we run stored procedures over the mysql driver,
we need to close the connection after executing the commands

        $this->_close();
    }

    return $members;
}

// note the "MODEL_" prefix in order to distinguish this
```



count\_members from the other functions named alike that may occur

```
function MODEL_count_members() {
    $count = 0; // initialize "count" variable

    if($this->_connect() === true){
        // note the "MYSQL_" prefix in order to distinguish this
count_members from the other functions named alike that may occur

        $sql = "CALL MYSQL_count_members()";
        $query = $this->db->query($sql);

        if($query->num_rows() > 0){
            $row = $query->result_array();
            $count = $row[0]['count'];
        }

        $query->free_result();
        $this->_close();
    }

    return $count;
}
```

// note the "MODEL\_" prefix in order to distinguish this  
select\_member\_info from the other functions named alike that may occur

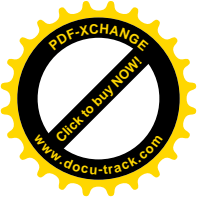
```
function MODEL_select_member_info() {
    $member = array(); // initialize "member" variable

    $this->_id = isset($this->_id)?$this->_id:0;

    if($this->_connect() === true){
        // note the "MYSQL_" prefix in order to distinguish this
select_member_info from the other functions named alike that may occur

        $sql = "CALL MYSQL_select_member_info( ".$this->_id." )";
        $query = $this->db->query($sql);

        if($query->num_rows() > 0){
            foreach($query->result_array() as $row){
                array_push($member, $row);
            }
        }
    }
}
```



```
        $query->free_result();
        $this->_close();
    }

    return $member;
}

function _connect() {
    $this->load->database('default', FALSE);

    if(isset($this->db->conn_id)) {
        return true;
    } else {
        return false;
    }
}

function _close() {
    $this->db->close($this->db->conn_id);
}
}
?>
```

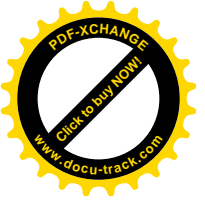
Step 4. Create a Nusoap webservice named "MemberWSVC.php":

```
<?php
class MemberWSVC extends Controller {
    function __construct() {
        parent::Controller();

        $this->load->library("Nusoap_lib");
        $this->load->model("Member");

        $this->nusoap_server = new soap_server();
        $this->nusoap_server->configureWSDL("MemberWSDL",
"urn:MemberWSDL");

        $this->nusoap_server->wsdl->addComplexType(
            "MembersRecordset",
            "complexType",
            "array",
            "",
            "SOAP-ENC:Array",
            array(
                "id"=>array("name"=>"id", "type"=>"xsd:int"),
```

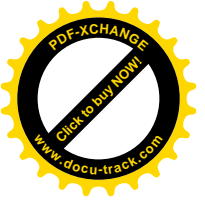


```
                "firstname"=>array("name"=>"firstname",
"type"=>"xsd:string"),
                "lastname"=>array("name"=>"lastname",
"type"=>"xsd:string"),
                "birthday"=>array("name"=>"birthday",
"type"=>"xsd:date")
            )
        );

$this->nusoap_server->wsdl->addComplexType(
    "MemberRecordset",
    "complexType",
    "array",
    "",
    "SOAP-ENC:Array",
    array(
        "id"=>array("name"=>"id", "type"=>"xsd:int"),
        "firstname"=>array("name"=>"firstname",
"type"=>"xsd:string"),
        "lastname"=>array("name"=>"lastname",
"type"=>"xsd:string"),
        "birthday"=>array("name"=>"birthday",
"type"=>"xsd:date")
    )
);

$this->nusoap_server->register(
    "selectMemberInfo",
    array(
        "id" => "xsd:int"
    ),
    array("return"=>"tns:MemberRecordset"),
    "urn:MemberWSDL",
    "urn:MemberWSDL#selectMemberInfo",
    "rpc",
    "encoded",
    "Retrieves member's info'"
);

$this->nusoap_server->register(
    "selectMembers",
    array(
        "start" => "xsd:int",
        "limit" => "xsd:int",
        "sortField" => "xsd:string",
```



```
        "sortOrder" => "xsd:string"
    ),
    array("return"=>"tns:MembersRecordset"),
    "urn:MemberWSDL",
    "urn:MemberWSDL#selectMembers",
    "rpc",
    "encoded",
    "Retrieves members' list"
);

$this->nusoap_server->register(
    "countMembers",
    array(),
    array("return"=>"xsd:int"),
    "urn:MemberWSDL",
    "urn:MemberWSDL#countMembers",
    "rpc",
    "encoded",
    "Counts members"
);
}

function index() {
    if($this->uri->rsegment(3) == "wsdl") {
        $_SERVER['QUERY_STRING'] = "wsdl";
    } else {
        $_SERVER['QUERY_STRING'] = "";
    }

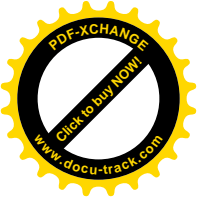
    $this->nusoap_server->
>service(file_get_contents("php://input"));
}

// note the "WS_" prefix (meaning Webservice) in order to
distinguish this select_members from the other functions named alike
that may occur

function WS_select_members() {
    function selectMembers($start, $limit, $sortField, $sortOrder)
    {

        $CI =& get_instance();

        $CI->Member->_start = $start;
        $CI->Member->_limit = $limit;
        $CI->Member->_sortField = $sortField;
```



```
        $CI->Member->_sortOrder = $sortOrder;

        $rows = $CI->Member->MODEL_select_members();

        return $rows;
    }

    $this->nusoap_server->
>service(file_get_contents("php://input"));
    }

    // note the "WS_" prefix (meaning Webservice) in order to
distinguish this select_member_info from the other functions named
alike that may occur

    function WS_select_member_info() {
        function selectMemberInfo($id) {
            $CI =& get_instance();

            $CI->Member->_id = $id;

            $row = $CI->Member->MODEL_select_member_info();

            return $row;
        }

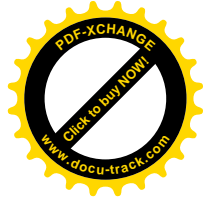
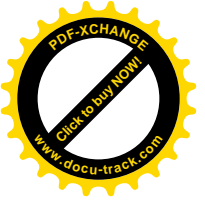
        $this->nusoap_server->
>service(file_get_contents("php://input"));
    }

    function WS_count_members() {
        function countMembers($id) {
            $CI =& get_instance();

            $value = $CI->Member->MODEL_count_members();

            return $value;
        }

        $this->nusoap_server->
>service(file_get_contents("php://input"));
    }
}
?>
```



Step 5. Create a controller named "member\_client.php":

```
<?php
class Member_Client extends Controller {

    function __construct() {
        parent::Controller();

        // load the needed libraries

        $this->load->library('Xajax_lib');
        $this->load->library('Nusoap_lib');

        // instantiate the xajax object

        $this->xajax = new xajax();

        // register the xajax functions

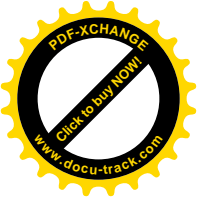
        $this->xajax->registerFunction(array('select_member_info',
&$this, 'CTRL_select_member_info'));
        $this->xajax->registerFunction(array('select_members', &$this,
'CTRL_select_members'));
        $this->xajax->processRequest();
    }

    function index() {
        $id = 0; // initialize "id" variable
        $start = 0; // initialize "start" variable
        $limit = 5; // initialize "limit" variable
        $sortField = "lastname"; // initialize "sortField" variable
        $sortOrder = "ASC"; // initialize "sortOrder"
variable

        // assign the variables to the view

        $this->view->set('id', $id);
        $this->view->set('start', $start);
        $this->view->set('limit', $limit);
        $this->view->set('sortField', $sortField);
        $this->view->set('sortOrder', $sortOrder);

        // load the member_client view
```



```
        $this->view->load('member_client');
    }

    // note the "CTRL_" prefix (meaning CONTROLLER) in order to
    // distinguish this select_members from the other functions named alike
    // that may occur

    function CTRL_select_members($params) {
        $objResponse = new xajaxResponse();

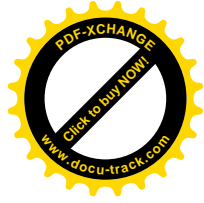
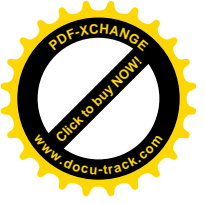
        $members_list = array();
        $members_count = 0;

        $this->soapclient = new
nusoapclient('http://localhost/index.php/MemberWSVC/WS_select_members/w
sdl');

        if(!$this->soapclient->fault)
        {
            if(!$this->soapclient->getError())
            {
                $members_list = $this->soapclient->call(
                    'selectMembers',
                    array($params['start'], $params['limit'],
$params['sortField'], $params['sortOrder']),
                    'urn:MemberWSDL',
                    'urn:MemberWSDL#selectMembers'
                );
            }
        }

        $this->soapclient = new
nusoapclient('http://localhost/index.php/MemberWSVC/WS_count_members/ws
dl');

        if(!$this->soapclient->fault)
        {
            if(!$this->soapclient->getError())
            {
                $members_count = $this->soapclient->call(
                    'countMembers',
                    array(),
                    'urn:MemberWSDL',
                    'urn:MemberWSDL#countMembers'
                );
            }
        }
    }
}
```



```
    }
}

$script = "members_count = ".$members_count."; \r\n";
$script .= "members_page = ".$params['page']."; \r\n";
$script .= "var members = []; \r\n";

foreach($members_list as $key => $value) {
    $script .= "members.push({id: ".$value['id']." , firstname:
escape('".$value['firstname']."' ), lastname:
escape('".$value['lastname']."' ), birthday:
'".$value['birthday']."' }); \r\n";
}

// note the "JS_" prefix (meaning Javascript) in order to
distinguish this select_members from the other functions named alike
that may occur

$script .= "JS_select_members(members, members_count); \r\n";

$objResponse->script($script);

return $objResponse;
}

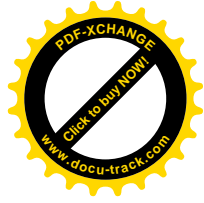
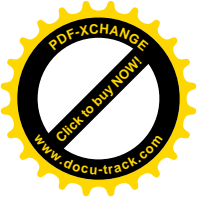
// note the "CTRL_" (meaning CONTROLLER) prefix in order to
distinguish this select_member_info from the other functions named
alike that may occur

function CTRL_select_member_info($id) {
    $objResponse = new xajaxResponse();

    $member = array();

    $this->soapclient = new
nusoapclient('http://localhost/index.php/MemberWSVC/WS_select_member_in
fo/wsd1');

    if(!$this->soapclient->fault)
    {
        if(!$this->soapclient->getError())
        {
            $member = $this->soapclient->call(
                'selectMemberInfo',
                array($id),
```



```
        'urn:MemberWSDL',
        'urn:MemberWSDL#selectMemberInfo'
    );
}
}

$script = "var member = [];\r\n";

foreach($member as $key => $value) {
    $script .= "member.push({id: ".$value['id'].", firstname:
escape('".$value['firstname']."), lastname:
escape('".$value['lastname']."), birthday:
'".$value['birthday']."});\r\n";
}

// note the "JS_" prefix (meaning Javascript) in order to
distinguish this select_members from the other functions named alike
that may occur

$script .= "JS_select_member_info(member);\r\n";

$objResponse->script($script);

return $objResponse;
}
?>
```

Step 6. Create a corresponding view named "member\_client.php":

```
<html>
<head>
<script type="text/javascript" src="/js/jquery-1.1.4.js"></script>
<?php isset($this->xajax)?$this->xajax->printJavascript('/js/'):""; ?>
<script type="text/javascript">
/* we need to pass the assigned variables from controller */
/* to the xajax registered function */
/*  */
var start = &lt;?php echo $start; ?&gt;;
var limit = &lt;?php echo $limit; ?&gt;;
var sortField = '&lt;?php echo $sortField; ?&gt;';
var sortOrder = '&lt;?php echo $sortOrder; ?&gt;';
var pages = 1;

// loader for members' list</pre></div>
```



```
function JS_members_loading() {
    $('#member_info').hide();
    $('#members_list').html('<div
id="members_loader">loading...</div>');
}

// loader for member's info

function JS_member_info_loading() {
    $('#member_info').show();
    $('#member_info_details').hide();
    $('#member_info_loader').show();
}

function JS_select_members(members, count) {
    var member_link = [];
    var page_link = [];

    $('#member_info').hide();

    if(members.length > 0) {
        $('#members_list').show();

        // show members count

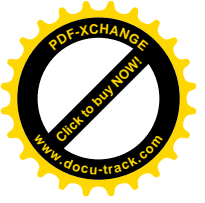
        $('#members_count').html(count + ' members');

        // build link for each page

        if(limit > 0) {
            pages = Math.ceil(count/limit);

            for(j = 0; j < pages; j++) {
                params[j] = new Object();
                params[j].page = j;
                params[j].start = j*limit;
                params[j].limit = limit;
                params[j].sortField = sortField;
                params[j].sortOrder = sortOrder;

                page_link[j] = '<a id="page_link_' + j + '"
href="javascript:void(0)" onclick="JS_members_loading();
ajax_select_members(params[' + j + '])" style="padding: 1px">' + (j +
1) + '</a> ';
            }
        }
    }
}
```



```
    }

    $('#members_pages').html(page_link.join(''));

    // select current page

    $('#page_link_' + members_page).css('background-color',
'#ff0000');
    }

    // build link for each member

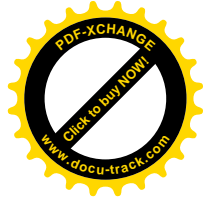
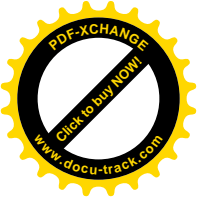
    for(i = 0; i < members.length; i++) {
        member_link[i] = '<a href="javascript:void(0)"
onclick="JS_member_info_loading(); xajax_select_member_info(' +
members[i].id + ')">' + members[i].lastname + ', ' +
members[i].firstname + '</a><br/>';
    }

    // complete members' list

    $('#members_list').html(member_link.join(''));
} else {
    $('#members_loader').html("there is no member in table");
}
}

function JS_select_member_info(member) {
    if(member.length > 0) {
        $('#member_info_loader').hide();
        $('#member_info_details').show();
        $('#member_id').html('ID: ' + member[0].id);
        $('#member_firstname').html('Firstname: ' +
member[0].firstname);
        $('#member_lastname').html('Lastname: ' + member[0].lastname);
        $('#member_birthday').html('Birthday: ' + member[0].birthday);
    }
}
/* ]]> */
</script>
</head>
<body>

<div style="float: left; padding: 3px; width: 150px">
    <div id="members_count"></div>
```



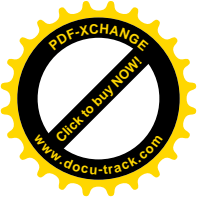
```
<div id="members_list" style="height: 100px">
    <div id="members_loader">loading...</div>
</div>
<div id="members_pages"></div>
</div>
<div style="float: left; padding: 3px; width: 150px">
    <div id="member_info" style="display: none; float: left; height:
100px; width: 150px; padding: 5px; border: 1px solid red">
        <div id="member_info_loader">loading...</div>
        <div id="member_info_details" style="display: none;">
            <div><b>Member Info</b></div>

            <div id="member_id"></div>
            <div id="member_firstname"></div>
            <div id="member_lastname"></div>
            <div id="member_birthday"></div>
        </div>
    </div>
</div>

<script type="text/javascript">
/* this script runs when we load the page for the first time */
/*  */
    var members_count = 0;
    var members_page = 0;

    params = new Object();
    params.page = members_page;
    params.start = start;
    params.limit = limit;
    params.sortField = sortField;
    params.sortOrder = sortOrder;
    xajax_select_members(params);
/* ]]&gt; */
&lt;/script&gt;

&lt;/body&gt;
&lt;/html&gt;</pre></div><div data-bbox="142 814 594 834" data-label="Text"><p>Step 7. Create a new route in the routes.php file:</p></div><div data-bbox="142 850 693 868" data-label="Text"><pre>$route['MemberWSVC/wsdl'] = "MemberWSVC/index/wsdl";</pre></div>
```



Step 8. Set the database driver to "mysqli" driver because we use mysql stored procedures.

Step 9. Place the above created files in the following folders:

"member.php" in application folder / models

"MemberWSVC.php" in your application folder / controllers

"member\_client.php" (the controller) in application folder / controllers

"member\_client.php" (the view ) in application folder / views

Step 10. Configure the mysql server to access the database and the table "members" correctly.

Step 11. Access the location:

[http://localhost/index.php/member\\_client/index](http://localhost/index.php/member_client/index) and if any of the above steps were completed successfully you should see a list of members splitted on three pages and when you click on a member a div containing the member's data should appear in page.