

```

<?php if ( ! defined('BASEPATH')) exit('no direct script access allowed');
class collector extends CI_Controller {
    public function __construct()
    {
        parent::__construct();
    }

    public function index()
    {
        error_reporting(0); // mematikan semua error yang terjadi
        while(1) {
            echo "----- check server ".date('F j, Y, g:i a').PHP_EOL;
            $server_list = $this->system_model->get_list_all_server(); //1 = ab 2 = c ip address
            //get service each server
            foreach($server_list as $server)
            {
                //1
                $service_list = $this->system_model->get_list_service_each_server($server->server_id);
                echo "---- Serverhost '$server->server_name.' status is ",($this->_checkhost($server->server_ipaddress) == 1 ? 'UP' :
'DOWN').PHP_EOL;
                if($this->_checkhost($server->server_ipaddress) == 1)
                {
                    foreach($service_list as $service)
                    {
                        echo $service->service_monitoring_id.' Service '.$service->service_name.' ON Server '.$server->server_name.'
('$server->server_ipaddress.') IS ',($this->_check_service($server->server_ipaddress,$service->service_port) == 1 ? 'UP' : 'DOWN').PHP_EOL;
                        $data = array (
                            'monitoring_id' => $service->service_monitoring_id,
                            'service_status' => ($this->_check_service($service->server_ipaddress,$service->service_port)
== 1 ? 'up' : 'down')
                        );
                        $this->system_model->service_log($data);
                    }
                }
                sleep(10);
            }
        }
    }
    public function get_service_status($server_id)
    {

```

```

        $service_list = $this->system_model->get_list_service_each_server($server_id); //mendapatkan list service yang sudah di daftarkan di
        foreach($service_list as $service)
        {
            $data = array (
                'monitoring_id' => $service->service_monitoring_id,
                'service_status' => ($this->_check_service($service->server_ipaddress,$service->service_port) == 1 ? 'up' :
'down')
            );
            $this->system_model->service_log($data);
        }
    }

    private function _checkhost($ipaddress)
    {
        $shoststatus = 1;
        $ping_exe = exec("ping -n 1 $ipaddress", $ping_result, $pr);
        if(count($ping_result) > 1)
        {
            $shoststatus = 1;
        }
        else
        {
            $shoststatus = 0;
        }
        return $shoststatus;
    }

    private function _check_service($ipaddress,$port)
    {
        $servicestatus = '';
        if ($sx = @fsockopen($ipaddress, $port, $errno, $errstr, 3)) // attempt to connect
        {
            $servicestatus = 0; //if 0 defined as host down
        }
        else
        {
            $servicestatus = 1;
            if ($sk)
            {
                @fclose($sk); //close connection
            }
        }
        return $servicestatus;
    }

```

Universitas Esa Unggul

L_2

```

private $pagetitle = 'Server Monitoring | ';
protected $hashkey = 'teguh123';

function __construct()
{
    parent::__construct();
    $this->load->library('grocery_CRUD');
}

function _check_session()
{
    if($this->session->userdata('islogged') == NULL)
    {
        return FALSE;
    }
    else
    {
        return TRUE;
    }
}

public function index()
{
    if($this->_check_session()== TRUE)
    {
        $data['page_title'] = $this->pagetitle.'Administration';
        $data['module'] = 'backend/backend';
        $data['customheader'] = 'backend_header_component';
        $this->load->view('master_view',$data);
    }
    else
    {
        $message = "Please Login to access Admin Page";
        if(isset($this->session->flashdata))
        {
            $message = $this->session->flashdata;
        }
        $data['css_component'] = 'login.css';
        $data['message'] = $message;
        $data['customheader'] = 'login_header_component';
        $data['page_title'] = $this->pagetitle.'Login Page';
        $data['module'] = 'login/login';
        $this->load->view('master_view',$data);
    }
}

function login()
{
    if($this->_check_session() == FALSE)
    {
        $uname = $this->input->post('username');
        $pass = $this->input->post('password');
        if(($uname != '') && ($pass != ''))
        {
            $status = $this->system_model->check_login($uname,$pass);
            if($status == TRUE)
            {
                $sesdata = array(
                    'username' => $this->encrypt->encode($this->input->post('username'),$this->hashkey),
                    'user' => $this->system_model->get_data_user($uname)->username,
                    'islogged' => TRUE
                );
                $this->session->set_userdata($sesdata);
                redirect('backend/config', 'refresh');
            }
            else
            {
                $this->session->sess_destroy();
                $this->session->flashdata('Incorrect username / Password');
                redirect('backend', 'refresh');
            }
        }
        else
        {
            $this->session->flashdata('Please Insert Username & Password');
            redirect('backend', 'refresh');
        }
    }
    else
    {
        redirect('backend', 'refresh');
    }
}

function logout()
{
    $this->session->sess_destroy();
    $this->session->flashdata('You Are Logout');
    redirect('backend', 'refresh');
}

```

```

<?php
class System_model extends CI_Model {
    protected $hashkey = 'teguh123';

    function check_login($name,$pass)
    {
        $this->db->where('username',$name);
        $this->db->where('isactive','Y');
        $encodepass = $this->db->get('system_user',1,0)->row()->password;
        $spassword = $this->encrypt->decode($encodepass,$this->hashkey);

        if($pass == $spassword)
        {
            //set online
            $data = array('isonline' => 'Y');
            $this->db->where('username', $name);
            $this->db->update('system_user',$data);
            return TRUE;
        }
        else
        {
            $data = array('isonline' => 'N');
            $this->db->where('username', $name);
            $this->db->update('system_user',$data);
            return FALSE;
        }
    }

    function get_data_user($username)
    {
        $this->db->where('username', $username);
        return $this->db->get('system_user',1,0)->row();
    }

    function hardware_log($data)
    {
        $this->db->insert('hardware_history',$data);
    }

    function service_log($data)
    {
        $newdata = $this->check_last_service_state($data['monitoring_id'],$data['service_status']);
        if($newdata == 'change')
    }

    {
        $newdata = $this->check_last_service_state($data['monitoring_id'],$data['service_status']);
        if($newdata == 'change')
        {
            $this->db->insert('service_history',$data);
            $last_id = $this->db->insert_id();
            $this->send_notification('service',$last_id);
            return 1;
        }
        else
        {
            return 0;
        }
    }

    function check_last_service_state($monitoring_id,$service_status)
    {
        error_reporting(0);
        $datastatus = '';
        $sql = "SELECT service_status FROM service_history WHERE monitoring_id = $monitoring_id ORDER BY timestamp DESC LIMIT 0,1";
        $last_state = $this->db->query($sql)->row()->service_status;
        if($service_status == $last_state)
        {
            $datastatus = 'notchange';
        }
        else
        {
            $datastatus = 'change';
        }
        return $datastatus;
    }

    function get_location()
    {
        return $this->db->get('master_location')->result();
    }

    function get_location_detail($location_id)
    {
        $this->db->where('location_id',$location_id);
        return $this->db->get('master_location',1,0)->row();
    }
}

```