

## SOURCE CODE SIMAK

### A. BACK END

#### 1. Users

##### a. Controller

```
<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

use Restserver\Libraries\REST_Controller;

require APPPATH . '/libraries/REST_Controller.php';

class Users extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('User_model', 'UserModel');
    }

    public function changepassword_post()
    {
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
            $Output = $this->UserModel->ChangesPassword($_POST,
$is_valid_token['data']->id);
            if (!empty($Output) && $Output != false) {
                $message = [
                    'status' => true,
                    'message' => "Changes Password Success",
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            }else{
                $message = [
                    'status' => false,
                    'message' => "Password Lama Anda Tidak Sesuai",
                ];
            }
        }
    }
}
```

```

];
$this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}
}else{
$message = [
'status' => false,
'message' => "Anda tidak memiliki Akses",
];
$this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}
}

public function changeuser_post()
{
$this->load->library('Authorization_Token');
$is_valid_token = $this->authorization_token->validateToken();
if ($is_valid_token['status'] === true) {
$_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
$Output = $this->UserModel->ChangesUsername($_POST,
$is_valid_token['data']->id);
if (!empty($Output) && $Output != false) {
$message = [
'status' => true,
'message' => "Changes Username Success"
];
$this->response($message, REST_Controller::HTTP_OK);
}else{
$message = [
'status' => false,
'message' => "Gagal, Hubungi Admin"
];
$this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}
}else{
$message = [
'status' => false,
'message' => "Anda tidak memiliki Akses"
];
$this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}
}
}

```

```

/**
 * Add New User
 * @method : POST
 */
public function register_post()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if($is_valid_token['status']===true){
        $data = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        $Output = $this->UserModel->insert_user($data);
        if($Output['status']){
            $message = [
                'message' => "User Berhasil di Simpan",
                'data' => $Output['id']
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [
                'message' => "User Gagal di Simpan",
                'data' => $Output['id']
            ];
            $this->response($message,
REST_Controller::HTTP_BAD_REQUEST);
        }
    }else{
        $message = [
            'message' => "User Berhasil di Simpan",
            'data' => $Output['id']
        ];
        $this->response($message,
REST_Controller::HTTP_UNAUTHORIZED);
    }
}
}

```

```

public function login_post()
{
    $_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
    $this->form_validation->set_rules('Username', 'Username',
'trim|required');
}

```

```

        $this->form_validation->set_rules('Password',
        'trim|required|max_length[100]');
        if ($this->form_validation->run() == false) {
            $message = array(
                'status' => false,
                'error' => $this->form_validation->error_array(),
                'message' => validation_errors(),
            );
            $this->response($message,
            REST_Controller::HTTP_NOT_FOUND);
        } else {
            $Output = $this->UserModel->user_login($this->input-
            >post('Username'), $this->input->post('Password'));
            if (!empty($Output && $Output != false)) {
                $this->load->library('Authorization_Token');
                $token_data['id'] = $Output->Id;
                $token_data['Username'] = $Output->Username;
                $token_data['Email'] = $Output->Email;
                $token_data['NamaUser'] = $Output->NamaUser;
                $token_data['RoleUser'] = $Output->role->Role;
                $token_data['time'] = time();
                $UserToken = $this->authorization_token-
                >generateToken($token_data);
                $return_data = [
                    'IdUser' => $Output->Id,
                    'Username' => $Output->Username,
                    'Email' => $Output->Email,
                    'NamaUser' => $Output->NamaUser,
                    'RoleUser' => $Output->role,
                    'Token' => $UserToken
                ];
                $message = [
                    'status' => true,
                    'data' => $return_data,
                    'message' => "Login Berhasil",
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            } else {
                $message = [
                    'status' => false,
                    'message' => "Username anda tidak ditemukan Periksa
                Username dan Password anda",
                ];
                $this->response($message,
                REST_Controller::HTTP_NOT_FOUND);
            }
        }
    }

```

```

    }
}

public function GetBiodata_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $Output = $this->UserModel->GetBiodata($is_valid_token['data']);
        if (!empty($Output) && $Output != false) {
            $message = [
                'status' => true,
                'data' => $Output
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

public function AmbilUser_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $Output = $this->UserModel->Select();
        if (!empty($Output) && $Output != false) {
            $message = [
                'status' => true,
                'data' => $Output
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

public function ResetPassword_put()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $_POST = json_decode($this->security->xss_clean($this->input->raw_input_stream), true);
        $Output = $this->UserModel->Reset($_POST);
        if ($Output) {
            $message = [
                'status' => true,
                'message'=> "anda berhasil melakukan reset password"
            ];

```

```

        $this->response($message, REST_Controller::HTTP_OK);
    }else{
        $message = [
            'status' => false,
        ];
        $this->response($message, REST_Controller::HTTP_OK);
    }
}
}
public function getRole_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $Output = $this->UserModel->getRole();
        if ($Output) {
            $message = [
                'status' => true,
                'data'=> $Output
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [
                'status' => false,
                'data'=> []
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}
}
public function changeUserInRole_post()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        $Output = $this->UserModel->ChangeUserRole($_POST);
        if ($Output) {
            $message = [
                'status' => true,
                'message'=> "Proses Berhasil"
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [

```

```

        'status' => false,
        'message'=> "Proses Gagal"
    ];
    $this->response($message,
REST_Controller::HTTP_BAD_REQUEST);
    }
}
}
public function userUpdate_put()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        $Output = $this->UserModel->userUpdate($_POST);
        if ($Output) {
            $message = [
                'status' => true,
                'message'=> "Proses Berhasil"
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [
                'status' => false,
                'message'=> "Proses Gagal"
            ];
            $this->response($message,
REST_Controller::HTTP_BAD_REQUEST);
        }
    }
}
}
}
}

```

## b. Model

```
<?php
```

```

class User_Model extends CI_Model
{
    protected $UserTable = 'user';
    protected $MahasiswaTable = 'mahasiswa';
    protected $UserinRoleTable = 'userinrole';
    protected $RoleTable = 'role';
    protected $PegawaiTable = 'pegawai';
    public function insert_user($UserData)
    {

```

```

$username = $UserData['Username'];
$result = $this->db->query(
    "SELECT * FROM user WHERE Username = '$username'"
);
if($result->num_rows()>0){
    $message = [
        'status' => false,
        'id' => $result->row('Id')
    ];
    return $message;
}else{
    $UserData['Password'] = md5($UserData['Password']);
    $this->db->insert($this->UserTable, $UserData);
    $message = [
        'status' => true,
        'id' => $result->row('Id')
    ];
    return $message;
}
}

```

```

public function ChangesPassword($data, $Id)
{
    $this->db->where('Id', $Id);
    $this->db->where('Password', md5($data['OldPassword']));
    $resultCek = $this->db->get('user');
    if ($resultCek->num_rows()) {
        $this->db->set('Password', md5($data['NewPassword']));
        $this->db->where('Id', $Id);
        $result = $this->db->update('user');
        return true;
    } else {
        return false;
    }
}

```

```

public function ChangesUsername($data, $Id)
{
    $this->db->set('Username', $data['Username']);
    $this->db->where('Id', $Id);
    $result = $this->db->update('user');
    if($result){
        return true;
    }else{
        return false;
    }
}

```



```

    }

    public function fetch_all_users()
    {
        $query = $this->db->get('user');
        foreach ($query->result() as $row) {
            $user_data[] = [
                'Username' => $row->Username,
                'Email' => $row->Email,
                'Insert' => $row->Insert,
                'Update' => $row->Update,
            ];
        }
        return $user_data;
    }

    public function user_login($Username, $Password)
    {
        $Pass = md5($Password);
        $this->db->select("user.Id, user.Username, user.Password,
user.Email, role>Nama as RoleName");
        $this->db->join('userinrole', 'userinrole.IdUser = user.Id', 'left');
        $this->db->join('role', 'role.Id = userinrole.RoleId', 'left');
        $this->db->where('Email', $Username);
        $this->db->or_where('Username', $Username);
        $this->db->where('Password', $Pass);
        $this->db->where('Status', true);
        $q = $this->db->get($this->UserTable);

        if ($q->num_rows()) {
            $a = $q->row();
            $Id = $q->row('Id');
            $this->db->select('*');
            $this->db->join('role', 'role.Id=userinrole.RoleId', 'left');
            $this->db->where('IdUser', $q->row('Id'));
            $roleinuser = $this->db->get($this->UserinRoleTable);
            $Tampung = $roleinuser->result_array();
            if ($roleinuser->num_rows()) {
                $this->db->where('Id', $roleinuser->row('RoleId'));
                $role = $this->db->get($this->RoleTable);
                $datarole = $role->row('Nama');
                if ($datarole == 'Mahasiswa') {
                    $this->db->where('IdUser', $q->row('Id'));
                    $Biodata = $this->db->get($this->MahasiswaTable);
                    if ($Biodata->num_rows()) {
                        $roleitem = array('Role' => array());
                    }
                }
            }
        }
    }

```

```

        $item = array('Nama' => $role->row('Nama'));
        array_push($roleitem['Role'], $item);
        $Nama = "NamaUser";
        $Role = "role";
        $a->$Nama = $Biodata->row('nmmhs');
        $a->$Role = (object) $roleitem;
    }
} else if($datarole == 'AdminPenggunaLain'){
    $this->db->where('IdUser', $q->row('Id'));
    $Biodata = $this->db->get("PenggunaLain");
    if ($Biodata->num_rows()) {
        $roleitem = array('Role' => array());
        foreach ($Tampung as &$value) {
            $item = array('Nama' => $value['Nama']);
            array_push($roleitem['Role'], $item);
        }
        $Nama = "NamaUser";
        $Role = "role";
        $a->$Nama = $Biodata->row('Nama');
        $a->$Role = (object) $roleitem;
    }
}
else {
    $this->db->where('IdUser', $q->row('Id'));
    $Biodata = $this->db->get($this->PegawaiTable);
    if ($Biodata->num_rows()) {
        $roleitem = array('Role' => array());
        foreach ($Tampung as &$value) {
            $item = array('Nama' => $value['Nama']);
            array_push($roleitem['Role'], $item);
        }
        $Nama = "NamaUser";
        $Role = "role";
        $a->$Nama = $Biodata->row('Nama');
        $a->$Role = (object) $roleitem;
    }
}
}
return $a;
} else {
    return false;
}
}
public function GetBiodata($data)
{

```

```

$result = $this->db->query("
    SELECT
        `role`.`Nama`
    FROM
        `user`
    LEFT JOIN `userinrole` ON `user`.`Id` = `userinrole`.`IdUser`
    LEFT JOIN `role` ON `role`.`Id` = `userinrole`.`RoleId`
    WHERE
        `user`.`Id` = '$data->id'
");
if($result->num_rows()>0){
    $dataresult = $result->result_object();
    $a = $dataresult[0];
    if($a->Nama == "Mahasiswa"){
        $resultMahasiswa = $this->db->query("
            SELECT
                *
            FROM
                `mahasiswa`
            WHERE IdUser = '$data->id'
        ");
        return $resultMahasiswa->result_array();
    }else if($a->Nama == "AdminPenggunaLain"){
        $resultMahasiswa = $this->db->query("
            SELECT
                *
            FROM
                `PenggunaLain`
            WHERE IdUser = '$data->id'
        ");
        return $resultMahasiswa->result_array();
    }else{
        $resultMahasiswa = $this->db->query("
            SELECT
                *
            FROM
                `pegawai`
            WHERE IdUser = '$data->id'
        ");
        return $resultMahasiswa->result_array();
    }
}
}
}
}

public function Select()

```

```

{
    $result = $this->db->query(
        "SELECT
            `pegawai`.*,
            `user`.`Status` AS `userStatus`,
            `user`.`Username`
        FROM
            `pegawai`
            LEFT JOIN `user` ON `user`.`Id` = `pegawai`.`IdUser`"
    );
    $dataPegawai = $result->result_object();
    foreach ($dataPegawai as $key => $value) {
        $resultUser = $this->db->query(
            "SELECT
                `userinrole`.*,
                `role`.*
            FROM
                `user`
                RIGHT JOIN `userinrole` ON `userinrole`.`IdUser` = `user`.`Id`
                LEFT JOIN `role` ON `role`.`Id` = `userinrole`.`RoleId`
            WHERE user.Id='$value->IdUser'"
        );
        $value->Role = $resultUser->result_object();
    }
    return $dataPegawai;
}

public function Reset($data)
{
    $IdUser = $data['IdUser'];
    $pass = md5("stimik1011");
    $this->db->set("Password", $pass);
    $this->db->where("Id", $IdUser);
    $result = $this->db->update("user");
    return $result;
}

public function userUpdate($data)
{
    $IdUser = $data['IdUser'];
    $Status;
    if($data['Status']==true){
        $Status = "true";
    }else{
        $Status = "false";
    }
    $this->db->set("Status", $Status);
    $this->db->where("Id", $IdUser);

```

```

        $result = $this->db->update("user");
        return $result;
    }
    public function getRole()
    {
        $result = $this->db->get("role");
        return $result->result_array();
    }
    public function ChangeUserRole($data)
    {
        if($data['status']==true){
            $this->db->where("IdUser", $data['IdUser']);
            $this->db->where("RoleId", $data['Id']);
            $num = $this->db->get("userinrole");
            if($num->num_rows()==0){
                $item=
                [
                    'RoleId' => $data['Id'],
                    'IdUser' => $data['IdUser']
                ];
                $result = $this->db->insert("userinrole", $item);
                return $result;
            }
        }else{
            $this->db->where("IdUser", $data['IdUser']);
            $this->db->where("RoleId", $data['Id']);
            $result = $this->db->delete('userinrole');
            return $result;
        }
    }
}

```

## 2. Dosen

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

```

```

use Restserver\Libraries\REST_Controller;

```

```

require APPPATH . '/libraries/REST_Controller.php';

```

```

class Dosen extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
    }
}

```

```

        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Dosen_model', 'DosenModel');
    }

```

```

public function GetDosen_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $result = $this->DosenModel->get();
        $message = [
            'status' => true,
            'data' => $result,
            'message' => "Success",
        ];
        $this->response($message, REST_Controller::HTTP_OK);
    }
}

```

## b. Model

```
<?php
```

```

class Dosen_Model extends CI_Model
{
    public function get()
    {
        $result = $this->db->query("
            SELECT
                *
            FROM
                `dosen`
            RIGHT JOIN `pegawai` ON `dosen`.`idpegawai` =
`pegawai`.`idpegawai`
            WHERE
                `dosen`.nidn!='null' AND
                `dosen`.scholarId!='null'
        ");
        $Data = $result->result_array();
        $b=array(
            "Data"=>array()
        );
    }
}

```

```

        foreach ($Data as $ke => $value) {
            $a = "http://cse.bth.se/~fer/googlescholar-
api/googlescholar.php?user=".$value['scholarId'];
            $response = $this->callAPI("GET", $a, false);
            if(strpos($response, "total_citations":, ')){
                $response = str_replace("total_citations":, "total_citations":
0',$response);
            }
            $data = array(
                "nidn"=>$value["nidn"],
                "nmsdn"=>$value["nmsdn"],
                "pendidikan"=>$value['pendakhir'],
                "scholarId" => $value['scholarId'],
                "Publikasi"=> $response
            );
            array_push($b["Data"], $data);
        }
        return $b["Data"];
    }
}

```

### 3. Mahasiswa

#### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

use Restserver\Libraries\REST_Controller;

require APPPATH . '/libraries/REST_Controller.php';

class Mahasiswa extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Mahasiswa_model', 'MahasiswaModel');
    }
    public function GetMahasiswa_get()
    {
        $npm = $this->get('npm');
        $Output = $this->MahasiswaModel->AmbilMahasiswa($npm);
    }
}

```

```

        if (!empty($Output)) {
            $message = [
                'status' => true,
                'data' => $Output['data'],
                'message' => "Success!",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        } else {
            $message = [
                'status' => false,
                'data' => [],
                'message' => "Kosong",
            ];
            $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
        }
    }
}

```

#### **b. Model**

```

<?php
class Mahasiswa_Model extends CI_Model
{
    public function AmbilMahasiswa($npm)
    {
        if ($npm != null && $npm!="undefined") {
            $ResultMahasiswa = $this->db->query("
            SELECT
                `mahasiswa`.`npm`,
                `mahasiswa`.`kdps`,
                `mahasiswa`.`jenjang`,
                `mahasiswa`.`kelas`,
                `mahasiswa`.`nmmhs`,
                `mahasiswa`.`tmlhr`,
                `mahasiswa`.`tglhr`,
                `mahasiswa`.`jk`,
                `mahasiswa`.`agama`,
                `mahasiswa`.`kewarga`,
                `mahasiswa`.`pendidikan`,
                `mahasiswa`.`nmsmu`,
                `mahasiswa`.`jursmu`,
                `mahasiswa`.`kotasmu`,
                `mahasiswa`.`kabsmu`,
                `mahasiswa`.`provsmu`,
                `mahasiswa`.`pekerjaan`,
                `mahasiswa`.`almt`,
                `mahasiswa`.`notlp`,

```



```

        `mahasiswa`.`status`,
        `mahasiswa`.`jmsaudara`,
        `mahasiswa`.`nmayah`,
        `mahasiswa`.`almtayah`,
        `mahasiswa`.`nmibu`,
        `mahasiswa`.`sumbiaya`,
        `mahasiswa`.`statuskul`,
        `mahasiswa`.`tgdaftar`,
        `mahasiswa`.`kurikulum`
FROM
    `mahasiswa` WHERE npm='$npm' AND statuskul in('AKTIF',
'CUTI', 'TIDAK AKTIF', 'TRANSFER');
if ($ResultMahasiswa->num_rows()) {
    $data = [
        'status' => true,
        'data' => $ResultMahasiswa->result(),
        'message' => "Success",
    ];
    return $data;
} else {
    $data = [
        'status' => true,
        'data' => $ResultMahasiswa->result(),
        'message' => "Tidak Data Mahasiswa",
    ];
    return $data;
}
} else {
    $ResultMahasiswa = $this->db->query("
SELECT
    `mahasiswa`.`npm`,
    `mahasiswa`.`kdps`,
    `mahasiswa`.`jenjang`,
    `mahasiswa`.`kelas`,
    `mahasiswa`.`nmmhs`,
    `mahasiswa`.`tmlhr`,
    `mahasiswa`.`tglhr`,
    `mahasiswa`.`jk`,
    `mahasiswa`.`agama`,
    `mahasiswa`.`kewarga`,
    `mahasiswa`.`pendidikan`,
    `mahasiswa`.`nmsmu`,
    `mahasiswa`.`jursmu`,
    `mahasiswa`.`kotasmu`,
    `mahasiswa`.`kabsmu`,
    `mahasiswa`.`provsmu`,

```

```

        `mahasiswa`.`pekerjaan`,
        `mahasiswa`.`almt`,
        `mahasiswa`.`notlp`,
        `mahasiswa`.`status`,
        `mahasiswa`.`jmsaudara`,
        `mahasiswa`.`nmayah`,
        `mahasiswa`.`almtayah`,
        `mahasiswa`.`nmibu`,
        `mahasiswa`.`sumbiaya`,
        `mahasiswa`.`statuskul`,
        `mahasiswa`.`tgdaftar`,
        `mahasiswa`.`kurikulum`,
        `user`.`Status` AS `userStatus`,
        `user`.`Email`,
        `user`.`Password`,
        `user`.`Username`
FROM
    `mahasiswa`
LEFT JOIN `user` ON `user`.`Id` = `mahasiswa`.`IdUser`
WHERE statuskul in('AKTIF', 'CUTI', 'TIDAK AKTIF',
'TRANSFER');
if ($ResultMahasiswa->num_rows()) {
    $data = [
        'status' => true,
        'data' => $ResultMahasiswa->result(),
        'message' => "Success",
    ];
    return $data;
} else {
    $data = [
        'status' => true,
        'data' => $ResultMahasiswa->result(),
        'message' => "Tidak Data Mahasiswa",
    ];
    return $data;
}
}
}
}

```

#### 4. Jadwal

##### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

```

```

use Restserver\Libraries\REST_Controller;

```

```

require APPPATH . '/libraries/REST_Controller.php';

class Jadwal extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Jadwal_model', 'JadwalModel');
    }
    /**
     * Jadwal Kuliah
     * @method : GET
     */
    public function JadwalKuliah_get()
    {
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $npm = $this->get('npm');
            $kelas = $this->get('kelas');
            $Output = $this->JadwalModel->getJadwalKuliah($npm, $kelas);
            if ($Output > 0 && !empty($Output)) {
                $message = [
                    'status' => true,
                    'data' => $Output
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            }else{
                $message = [
                    'status' => false,
                    'message' => "Tidak Ada Data",
                ];
                $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
            }
        }
    }
}

```

```

/**
 * Jadwal Mengajar Dosen
 * @method : GET
 */
public function JadwalDosen_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        // $npm = $this->get('nidn');
        $Output = $this->JadwalModel-
>getJadwalDosen($is_valid_token['data']);
        if ($Output > 0 && !empty($Output)) {
            $message = [
                'status' => true,
                'data' => $Output
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [
                'status' => false,
                'message' => [],
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

```

```

/**
 * Semua Jadwal
 * @method : POST
 */

public function jadwalall_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $Output = $this->JadwalModel->getAllJadwal();
        if ($Output > 0 && !empty($Output)) {
            $message = [
                'status' => true,
                'data' => json_encode($Output),
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

```

```

    } else {
        $message = [
            'status' => false,
            'message' => "Tidak Ada Data",
        ];
        $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
    }
} else {
    $message = [
        'status' => false,
        'message' => "Sessiom Habis",
    ];
    $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}

}

public function jadwalMahasiswa_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $Output = $this->JadwalModel-
>getjadwalmahasiswa($is_valid_token['data']->id);
        if ($Output > 0 && !empty($Output)) {
            if ($Output['message'] == 'TemKrsn') {
                $Data kirim = array(
                    'data' => $Output,
                );
                $message = [
                    'status' => true,
                    'data' => $Data kirim,
                    'set' => $Output['message']
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            } else if ($Output['message'] == 'Jadwal') {
                if ($Output['status'] == true) {
                    $Data kirim = array(
                        'data' => $Output['Jadwal']
                    );
                    $message = [
                        'status' => true,
                        'data' => $Data kirim,
                        'set' => $Output['message'],
                    ];
                    $this->response($message, REST_Controller::HTTP_OK);
                }
            }
        }
    }
}

```

```

        'mahasiswa' => $Output['mahasiswa'],
        'semester' => $Output['semester']
    ];
    $this->response($message, REST_Controller::HTTP_OK);
} else {
    $message = [
        'status' => false,
        'data' => array(),
        'message' => "Data Jadwal Belum Tersedia Segera
Hubungi Bagian BAAK Untuk ketersediaan",
    ];
    $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}
} else if($Output['message'] == 'Krms') {
    $Data kirim = array(
        'status' => false,
        'data' => $Output,
    );
    $message = [
        'status' => true,
        'data' => $Data kirim,
        'set' => $Output['message']
    ];
    $this->response($message, REST_Controller::HTTP_OK);
} else if($Output['message'] == 'Daftar Ulang') {
    $Data kirim = array(
        'status' => false,
        'data' => $Output,
    );
    $message = [
        'status' => true,
        'data' => $Data kirim,
        'set' => $Output['message']
    ];
    $this->response($message, REST_Controller::HTTP_OK);
} else {
    $message = [
        'status' => false,
        'set' => $Output['message']
    ];
    $this->response($message, REST_Controller::HTTP_OK);
}
}
} else {
    $message = [

```

```

        'status' => false,
        'message' => "Tidak Ada Data",
    ];
    $this->response($message,
    REST_Controller::HTTP_NOT_FOUND);
    }
    } else {
        $message = [
            'status' => false,
            'message' => "Sessiom Habis",
        ];
        $this->response($message,
    REST_Controller::HTTP_NOT_FOUND);
    }

    }

    public function pengajuanKRS_post()
    {
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST");
        header("Access-Control-Max-Age: 3600");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        $Output = $this->JadwalModel->pengajuanKRS_post($_POST);
        if ($Output > 0 && !empty($Output)) {
            $message = [
                'status' => true,
                'message' => "Pengajuan KRS Anda Berhasil",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        } else {
            $message = [
                'status' => false,
                'message' => "Pengajuan KRS Anda Gagal",
            ];
            $this->response($message,
    REST_Controller::HTTP_NOT_FOUND);
        }
    }
}
}
}

```

#### **b. Model**

```
<?php
```

```

class Jadwal_Model extends CI_Model
{
    protected $JadwalTable = 'jadwal_kuliah';
    protected $TahunAkademikTable = 'tahun_akademik';
    protected $MahasiswaTable = 'mahasiswa';
    protected $TemKrsTable = 'tem_krs';
    protected $TemDetailKrsTable = 'tem_detail_krs';
    protected $KrsTable = 'krs';
    protected $DetailKrsTable = 'krs_detail';
    public function insert_user(array $UserData)
    {
        $this->db->insert($this->UserTable, $UserData);
        return $this->db->insert_id();
    }

    public function getJadwalDosen($data)
    {
        $result = $this->db->query("
            SELECT
                `dosen`.`nidn`,
                `jadwal_kuliah`.*,          COUNT(krs_detail.npm)          AS
jumlahMahasiswa

            FROM
                `jadwal_kuliah`
            LEFT JOIN krs_detail ON krs_detail.thakademik =
jadwal_kuliah.thakademik AND
                krs_detail.gg = jadwal_kuliah.gg AND
                krs_detail.kmk = jadwal_kuliah.kmk AND
                krs_detail.kelas = jadwal_kuliah.kelas
            RIGHT JOIN `tahun_akademik` ON
`tahun_akademik`.`thakademik` =
                `jadwal_kuliah`.`thakademik` AND `tahun_akademik`.`gg` =
                `jadwal_kuliah`.`gg`
            LEFT JOIN `dosen_pengampu` ON
`dosen_pengampu`.`thakademik` =
                `tahun_akademik`.`thakademik` AND `dosen_pengampu`.`gg` =
                `tahun_akademik`.`gg` AND `jadwal_kuliah`.`idpengampu` =
                `dosen_pengampu`.`idpengampu`
            LEFT JOIN `dosen` ON `dosen`.`iddosen` =
`dosen_pengampu`.`iddosen`
            RIGHT JOIN `pegawai` ON `pegawai`.`idpegawai` =
`dosen`.`idpegawai`
            LEFT JOIN `matakuliah` ON `matakuliah`.`idmatakuliah` =
                `dosen_pengampu`.`idmatakuliah` AND matakuliah.kdps =
jadwal_kuliah.kdps
        ");
    }
}

```



WHERE

```
        `tahun_akademik`.`status` = 'AKTIF' AND
        `pegawai`.`idUser` = '$data->id' AND
        `dosen_pengampu`.`mengajar` = 'Y'
    GROUP BY      jadwal_kuliah.thakademik,      jadwal_kuliah.gg,
jadwal_kuliah.kelas, jadwal_kuliah.kmk
    ");
    return $result->result_object();
}
```

```
public function getJadwalKuliah($npm, $kelas)
{
    $result = $this->db->query("
    SELECT
    *
    FROM
    `krsm_detail`
    LEFT JOIN `tahun_akademik` ON `krsm_detail`.`thakademik` =
        `tahun_akademik`.`thakademik`      AND      `krsm_detail`.`gg`      =
        `tahun_akademik`.`gg`
    RIGHT JOIN `jadwal_kuliah` ON `jadwal_kuliah`.`thakademik` =
        `krsm_detail`.`thakademik`      AND      `jadwal_kuliah`.`gg`      =
        `krsm_detail`.`gg` AND
        `jadwal_kuliah`.`kmk`      =      `krsm_detail`.`kmk`      AND
        `jadwal_kuliah`.`kelas` =
        `krsm_detail`.`kelas`
    LEFT JOIN `mahasiswa` ON `mahasiswa`.`npm` =
        `krsm_detail`.`npm`

        WHERE      mahasiswa.npm      =      '$npm'      AND
tahun_akademik.status='AKTIF'");
    if($result->num_rows()){
        return $result->result_object();
    }else{
        return 0;
    }
}
```

```
public function getAllJadwal(Type $var = null)
{
    $hasil = $this->db->query("
    SELECT
        `jadwal_kuliah`.*,
        `matakuliah`.`smt`,
        `matakuliah`.`kurikulum`,
```

```

        `program_studi`.`nmps`,
        `dosen`.`nmdsn`,
        `dosen_pengampu`.`nidn`
FROM
    `jadwal_kuliah`
    LEFT JOIN `tahun_akademik` ON `jadwal_kuliah`.`thakademik` =
        `tahun_akademik`.`thakademik` AND `jadwal_kuliah`.`gg` =
        `tahun_akademik`.`gg`
    LEFT JOIN `matakuliah` ON `matakuliah`.`kmk` =
        `jadwal_kuliah`.`kmk`
        AND `matakuliah`.`kdps` = `jadwal_kuliah`.`kdps`
    LEFT JOIN `program_studi` ON `jadwal_kuliah`.`kdps` =
        `program_studi`.`kdps`
    LEFT JOIN `dosen_pengampu` ON `jadwal_kuliah`.`kmk` =
        `dosen_pengampu`.`kmk`
    RIGHT JOIN `dosen` ON `dosen`.`nidn` =
        `dosen_pengampu`.`nidn`
WHERE
    `tahun_akademik`.`status` = 'AKTIF' AND
    `dosen_pengampu`.`mengajar` = 'Y'
ORDER BY
    `matakuliah`.`smt`
");
if($hasil->num_rows())
{
    return $hasil->result_object();
}
}

```

```

public function getjadwalmahasiswa($idUser)
{
    $this->db->select('*');
    $this->db->where('IdUser', $idUser);
    $Mahasiswa = $this->db->get($this->MahasiswaTable);
    $StatusTA = "AKTIF";
    foreach ($Mahasiswa->result() as $value) {
        $DataMahasiswa = $value;
    }
    $this->db->select('*');
    $this->db->where('status', $StatusTA);
    $DataTahunAkademik = $this->db->get($this->
>TahunAkademikTable);
    foreach ($DataTahunAkademik->result() as $value) {
        $DataTA = $value;
    }
    if ($Mahasiswa->num_rows()) {

```

```

if ($DataMahasiswa->statuskul == "TIDAK AKTIF" ||
    $DataMahasiswa->statuskul == "TRANSFER" ||
    $DataMahasiswa->statuskul == "CUTI") {

    $this->db->where('npm', $DataMahasiswa->npm);
    $this->db->where('thakademik', $DataTA->thakademik);
    $this->db->where('gg', $DataTA->gg);
    $resultDU = $this->db->get('daftar_ulang');
    $TgSistem = date('Y-m-d');
    if($resultDU->num_rows()){
        if($DataTA->tglReg <= $TgSistem){
            $tgl = date('Y-m-d');
            $tglmhs = $resultDU->row('last_reg');
            $Tanggal_sistem = strtotime($tgl);
            $TglReg = strtotime($tglmhs);
            if($Tanggal_sistem<=$TglReg){
                $this->db->where('npm', $DataMahasiswa->npm);
                $this->db->where('thakademik', $DataTA->thakademik);
                $this->db->where('gg', $DataTA->gg);
                $resultTemKrsm = $this->db->get($this-
>TemKrsmTable);
                if ($resultTemKrsm->num_rows()) {
                    $this->db->where('IdKrsm', $resultTemKrsm-
>row('Id'));
                    $resultDetailKrsm = $this->db->get($this-
>TemDetailKrsmTable);
                    $Ddatas = array(
                        'TemKrsm' => json_encode($resultTemKrsm-
>result()),
                        'TemDetailKrsm' => json_encode($resultDetailKrsm-
>result()),
                        'message' => "TemKrsm",
                    );
                    return $Ddatas;
                } else {
                    if ($DataTahunAkademik->num_rows()) {
                        $this->db->select('*');
                        $this->db->join('matakuliah', 'matakuliah.kmk =
jadwal_kuliah.kmk', 'left');
                        $this->db->where('jadwal_kuliah.kdps',
$DataMahasiswa->kdps);
                        $this->db->where('jadwal_kuliah.thakademik',
$DataTA->thakademik);
                        $this->db->where('jadwal_kuliah.gg', $DataTA->gg);
                        // $this->db->where('kelas', $DataMahasiswa-
>kelas);

```

```

$this->db->group_start();
$this->db->where('kurikulum', $DataMahasiswa-
>kurikulum);

$this->db->or_where('kurikulum', 'ALL');
$this->db->group_end();
$this->db->order_by('matakuliah.smt', 'ASC');
$itemJadwal = $this->db->get($this->JadwalTable);
if ($itemJadwal->num_rows()) {
    $DataJadwal = array(
        'Jadwal' => json_encode($itemJadwal-
>result()),

        'Kelas' => $DataMahasiswa->kelas,
        'message' => 'Jadwal',
        'status' => true,
        'streg' => true,
        'mahasiswa' => $DataMahasiswa,
        'semester' => $resultDU->num_rows(),
    );
    return $DataJadwal;
}else{
    $DataJadwal = array(
        'message' => 'Jadwal',
        'status' => false,
        'streg' => true
    );
    return $DataJadwal;
}
}
}
}else{
    $DataJadwal = array(
        'message' => 'BatasReg',
        'status' => true
    );
    return $DataJadwal;
}
}else{
    $DataJadwal = array(
        'message' => 'MulaiReg',
        'status' => true
    );
    return $DataJadwal;
}
}
}

```

```

        'message' => 'Daftar Ulang',
        'status' => false
    );
    return $DataJadwal;
}
} else {
    $this->db->where('npm', $DataMahasiswa->npm);
    $this->db->where('thakademik', $DataTA->thakademik);
    $this->db->where('gg', $DataTA->gg);
    $resultKrsm = $this->db->get($this->KrsmTable);

    if ($resultKrsm->num_rows()) {
        $resultDetailKrsm = $this->db->query("
        SELECT
        `krsm_detail`.*,
        `khsm_detail`.`nhuruf`,
        `khsm_detail`.`nxsk`,
        `khsm_detail`.`ket`
        FROM
        `krsm_detail`
        LEFT JOIN `khsm_detail` ON `khsm_detail`.`thakademik` =
        `krsm_detail`.`thakademik` AND `khsm_detail`.`gg` =
        `krsm_detail`.`gg` AND
        `khsm_detail`.`npm` = `krsm_detail`.`npm` AND
        `khsm_detail`.`krmk` =
        `krsm_detail`.`krmk`
        WHERE `krsm_detail`.`npm` = '$DataMahasiswa->npm' AND
        `krsm_detail`.`thakademik` = '$DataTA->thakademik' AND
        `krsm_detail`.`gg` = '$DataTA->gg'");
        $DataKrsm = array(
            'Krsm' => $resultKrsm->result(),
            'DetailKrsm' => $resultDetailKrsm->result(),
            'message' => 'Krsm',
        );
        return $DataKrsm;
    } else {
        $DataJadwal = array(
            'message' => 'Daftar Ulang',
            'status' => false
        );
        return $DataJadwal;
    }
}
}
}

```

```
}
```

## 5. KRSM

### a. Controller

```
<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

use Restserver\Libraries\REST_Controller;

require APPPATH . '/libraries/REST_Controller.php';

class Krsm extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Krsm_model', 'KrsmModel');
    }
    public function ambilTemkrsm_post()
    {
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        $Status = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        if ($is_valid_token['status'] === true) {
            $Output = $this->KrsmModel->GetTem($is_valid_token['data'],
$Status['status']);
            if ($Output > 0 && !empty($Output)) {
                $message = [
                    'status' => true,
                    'data' => json_encode($Output),
                    'message' => "Success!",
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            } else {
                $message = [
                    'status' => false,
                    'data' => json_encode(array()),
                    'message' => "Data Kosong",
                ];
            }
        }
    }
}
```

```

];
$this->response($message,
REST_Controller::HTTP_NOT_FOUND);
    }
}

public function approvedKrsm_put()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    $_PUT = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
    if ($is_valid_token['status'] === true) {
        $Output = $this->KrsmModel->UpdateTemKrsm($_PUT);
        if (!empty($Output && $Output != false)) {
            $message = [
                'status' => true,
                'message' => "Berhasil melakukan Approved"
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

```

```

public function InsertItem_post()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $item = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);

        $Output = $this->KrsmModel->InsertItemKrsm($item);
        if (!empty($Output && $Output != false)) {

        }
    }
}

```

```

public function HapusItem_delete()
{
    $this->load->library('Authorization_Token');

    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {

```

```

        $item = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);

```

```

        $Output = $this->KrmsModel->DeleteItemKRSM($item['Id']);
        if (!empty($Output && $Output != false)) {
            $message = [
                'status' => true,
                'message' => "Pengajuan KRS Anda Berhasil",
            ];
        }
    }
}

```

```

public function pengajuanKRS_post()
{
    $this->load->library('Authorization_Token');

    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $_POST = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);

```

```

        $Output = $this->KrmsModel->Inser_Pengajuan((object) $this-
>input->post('krsm'), (object) $this->input->post('DetailKrms'));
        if ($Output) {
            $message = [
                'status' => true,
                'message' => "Pengajuan KRS Anda Berhasil"
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        } else {
            $message = [
                'status' => false,
                'message' => "Pengajuan KRS Anda Gagal"
            ];
            $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
        }
    }
}

```

```

public function AmbilKrms_get()
{
    $data = $_GET;
    $Output = $this->KrmsModel->get($data);
    $message = [

```



```

        'status' => true,
        'data' => $Output
    ];
    $this->response($message, REST_Controller::HTTP_OK);
}
}

```

## b. Model

```
<?php
```

```

class Krsm_Model extends CI_Model
{
    protected $KrsmTable = 'tem_krsm';
    protected $KrsmDetailTabel = 'tem_detail_krsm';
    protected $DosenWaliTabel = 'dosen_wali';
    protected $ProgramStudiTabel = 'program_studi';
    protected $PengampuhTabel = 'dosen_pengampu';
    protected $BaakTabel = 'baak';

    public function insert_user(array $UserData)
    {
        $this->db->insert($this->UserTable, $UserData);
        return $this->db->insert_id();
    }

    public function get($data)
    {
        $thakademik = $data['thakademik'];
        $gg = $data['gg'];
        $result = $this->db->query("CALL GetKrsmMhs('$thakademik','$gg')");
        return $result->result_array();
    }

    public function fetch_all_jadwal($IdUser)
    {
        $this->db->select('*');
        $this->db->where('IdUser', $IdUser);
        $Mahasiswa = $this->db->get($this->MahasiswaTable);
        $StatusTA = "AKTIF";

        foreach ($Mahasiswa->result() as $value) {
            $DataMahasiswa = $value;
        }
        if ($Mahasiswa->num_rows()) {
            if ($DataMahasiswa->statuskul == "TIDAK AKTIF" ||
                $DataMahasiswa->statuskul == "TRANSFER" ||

```

```

        $DataMahasiswa->statuskul == "CUTI") {
        $this->db->select('*');
        $this->db->where('status', $StatusTA);
        $DataTahunAkademik = $this->db->get($this->TahunAkademikTable);

        foreach ($DataTahunAkademik->result() as $value) {
            $DataTA = $value;
        }
        // $this->db->where();
        if ($DataTahunAkademik->num_rows()) {
            $this->db->select('*');
            $this->db->join('matakuliah', 'matakuliah.kmk =
jadwal_kuliah.kmk', 'left');
            $this->db->where('jadwal_kuliah.kdps', $DataMahasiswa->kdps);

            $this->db->where('thakademik', $DataTA->thakademik);
            $this->db->where('gg', $DataTA->gg);
            $this->db->where('kelas', $DataMahasiswa->kelas);
            $this->db->group_start();
            $this->db->where('kurikulum', $DataMahasiswa->kurikulum);
            $this->db->or_where('kurikulum', 'ALL');
            $this->db->group_end();
            $this->db->order_by('matakuliah.smt', 'ASC');
            $DataJadwal = $this->db->get($this->JadwalTable);

        }
        return $DataJadwal->result();
    }
} else {
    return false;
}
}

```

```

public function GetTem($data, $status)
{
    $temKrsm;
    $histori=[];
    $this->db->where('status', 'AKTIF');
    $resultthakademik = $this->db->get('tahun_akademik');
    $thakademik = $resultthakademik->result_object();
    if ($status == "Keuangan") {
        $this->db->select("
        `tem_krsm`.`ld`,
        `tem_krsm`.`thakademik`,
        `tem_krsm`.`gg`,

```

```

        `tem_krsm`.`npm`,
        `tem_krsm`.`sms`,
        `tem_krsm`.`dsn_wali`,
        `tem_krsm`.`ketjur`,
        `tem_krsm`.`admakademik`,
        `tem_krsm`.`jmsks`,
        `tem_krsm`.`tgkrsm`,
        `tem_krsm`.`status`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
        $this->db->join("mahasiswa", "`mahasiswa`.`npm` =
`tem_krsm`.`npm`, "LEFT");
        $this->db->where('tem_krsm.status', $status);
        $this->db->where('thakademik', $thakademik[0]->thakademik);
        $this->db->where('gg', $thakademik[0]->gg);
        $temKrsm = $this->db->get('tem_krsm');

        $this->db->select("
        `krsm`.`ldKrsm` AS ld,
        `krsm`.`thakademik`,
        `krsm`.`gg`,
        `krsm`.`npm`,
        `krsm`.`sms`,
        `krsm`.`dsn_wali`,
        `krsm`.`ketjur`,
        `krsm`.`admakademik`,
        `krsm`.`jmsks`,
        `krsm`.`tgkrsm`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
        $this->db->join("mahasiswa", "`mahasiswa`.`npm` = `krsm`.`npm`,
"LEFT");
        $this->db->where('thakademik', $thakademik[0]->thakademik);
        $this->db->where('gg', $thakademik[0]->gg);
        $resultkrsm = $this->db->get('krsm');

        foreach ($resultkrsm->result_object() as $key => $value) {
            $value->status = "Finish";
            array_push($histori, $value);
        }

        $this->db->select("
        `tem_krsm`.`ld`,
        `tem_krsm`.`thakademik`,
        `tem_krsm`.`gg`,
        `tem_krsm`.`npm`,

```

```

        `tem_krsm`.`sms`,
        `tem_krsm`.`dsn_wali`,
        `tem_krsm`.`ketjur`,
        `tem_krsm`.`admakademik`,
        `tem_krsm`.`jmsks`,
        `tem_krsm`.`tgkrsm`,
        `tem_krsm`.`status`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
        $this->db->join("mahasiswa",          "`mahasiswa`.`npm`          =
`tem_krsm`.`npm`, "LEFT");
        $this->db->where_not_in('tem_krsm.status', $status);
        $this->db->where('thakademik', $thakademik[0]->thakademik);
        $this->db->where('gg', $thakademik[0]->gg);
        $resultkrsm = $this->db->get('tem_krsm');

        foreach ($resultkrsm->result_object() as $key => $value) {
            array_push($histori, $value);
        }
    } else if($status=="Dosen Wali"){
        $this->db->select("
        `tem_krsm`.`ld`,
        `tem_krsm`.`thakademik`,
        `tem_krsm`.`gg`,
        `tem_krsm`.`npm`,
        `tem_krsm`.`sms`,
        `tem_krsm`.`dsn_wali`,
        `tem_krsm`.`ketjur`,
        `tem_krsm`.`admakademik`,
        `tem_krsm`.`jmsks`,
        `tem_krsm`.`tgkrsm`,
        `tem_krsm`.`status`,
        `dosen`.`nmdsn`,
        `dosen`.`nidn`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
        $this->db->join('dosen_wali',          'dosen_wali.npm=tem_krsm.npm',
'left');
        $this->db->join('dosen', 'dosen.nidn=dosen_wali.nidn', 'right');
        $this->db->join('pegawai', 'pegawai.idpegawai=dosen.idpegawai',
'right');
        $this->db->join("mahasiswa",          "`mahasiswa`.`npm`          =
`tem_krsm`.`npm`, "LEFT");
        $this->db->where("`tem_krsm`.`status`, $status);
        $this->db->where('pegawai.IdUser', $data->id);
        $temKrsm = $this->db->get('tem_krsm');

```

```

// Get Histori
$statuss = array('Keuangan', 'Dosen Wali');
$this->db->select("
`tem_krsm`.`Id`,
`tem_krsm`.`thakademik`,
`tem_krsm`.`gg`,
`tem_krsm`.`npm`,
`tem_krsm`.`sms`,
`tem_krsm`.`dsn_wali`,
`tem_krsm`.`ketjur`,
`tem_krsm`.`admakademik`,
`tem_krsm`.`jmsks`,
`tem_krsm`.`tgkrsm`,
`tem_krsm`.`status`,
`dosen`.`nmdsn`,
`dosen`.`nidn`,
`mahasiswa`.`nmmhs`,
`mahasiswa`.`kelas`");
$this->db->join('dosen_wali', 'dosen_wali.npm=tem_krsm.npm',
'left');
$this->db->join('dosen', 'dosen.nidn=dosen_wali.nidn', 'right');
$this->db->join('pegawai', 'pegawai.idpegawai=dosen.idpegawai',
'right');
$this->db->join("mahasiswa", "`mahasiswa`.`npm` =
`tem_krsm`.`npm`, "LEFT");
$this->db->where_not_in("`tem_krsm`.`status`, $statuss);
$this->db->where('pegawai.IdUser', $data->id);
$resultkrsm = $this->db->get('tem_krsm');
foreach ($resultkrsm->result_object() as $key => $value) {
    array_push($histori, $value);
}

$this->db->select("
`krsm`.`IdKrsm` AS Id,
`krsm`.`thakademik`,
`krsm`.`gg`,
`krsm`.`npm`,
`krsm`.`sms`,
`krsm`.`dsn_wali`,
`krsm`.`ketjur`,
`krsm`.`admakademik`,
`krsm`.`jmsks`,
`krsm`.`tgkrsm`,
`dosen`.`nmdsn`,
`dosen`.`nidn`,

```

```

        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
    $this->db->join('dosen_wali', 'dosen_wali.npm=krsm.npm', 'left');
    $this->db->join('dosen', 'dosen.nidn=dosen_wali.nidn', 'right');
    $this->db->join('pegawai', 'pegawai.idpegawai=dosen.idpegawai',
'right');
    $this->db->join("mahasiswa", "`mahasiswa`.`npm` = `krsm`.`npm`",
"LEFT");
    $this->db->where('pegawai.IdUser', $data->id);
    $this->db->where('krsm.thakademik', $thakademik[0]-
>thakademik);
    $this->db->where('krsm.gg', $thakademik[0]->gg);
    $resultkrsm = $this->db->get('krsm');
    foreach ($resultkrsm->result_object() as $key => $value) {
        $value->status = "Finish";
        array_push($histori, $value);
    }

}

}else{
    $this->db->select("
        `tem_krsm`.`Id`,
        `tem_krsm`.`thakademik`,
        `tem_krsm`.`gg`,
        `tem_krsm`.`npm`,
        `tem_krsm`.`sms`,
        `tem_krsm`.`dsn_wali`,
        `tem_krsm`.`ketjur`,
        `tem_krsm`.`admakademik`,
        `tem_krsm`.`jmsks`,
        `tem_krsm`.`tgkrsm`,
        `tem_krsm`.`status`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
    $this->db->join("mahasiswa", "`mahasiswa`.`npm` =
`tem_krsm`.`npm`", "LEFT");
    $this->db->join('program_studi',
'program_studi.kdps=mahasiswa.kdps', 'RIGHT');
    $this->db->join('kaprodi', 'kaprodi.idprodi=program_studi.idprodi',
'right');
    $this->db->join('pegawai', 'pegawai.idpegawai=kaprodi.idpegawai',
'right');
    $this->db->where('kaprodi`.`status`, 'AKTIF');
    $this->db->where('tem_krsm`.`status`, $status);
    $this->db->where('pegawai.IdUser', $data->id);
    $temKrsm = $this->db->get('tem_krsm');

```

```

        $this->db->select("
        `krsm`.`IdKrsm` AS Id,
        `krsm`.`thakademik`,
        `krsm`.`gg`,
        `krsm`.`npm`,
        `krsm`.`sms`,
        `krsm`.`dsn_wali`,
        `krsm`.`ketjur`,
        `krsm`.`admakademik`,
        `krsm`.`jmsks`,
        `krsm`.`tgkrsm`,
        `program_studi`.`kaprodi`,
        `dosen`.`nidn`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`");
        $this->db->join("mahasiswa", "`mahasiswa`.`npm` = `krsm`.`npm`",
"LEFT");
        $this->db->join('program_studi',
'program_studi.kdps=mahasiswa.kdps', 'left');
        $this->db->join('kaprodi', 'kaprodi.idprodi=program_studi.idprodi',
'left');
        $this->db->join('pegawai', 'pegawai.idpegawai=kaprodi.idpegawai',
'left');
        $this->db->join('dosen', 'dosen.idpegawai=pegawai.idpegawai',
'left');
        $this->db->join('daftar_ulang', 'daftar_ulang.npm=krsm.npm AND
daftar_ulang.thakademik=krsm.thakademik AND
daftar_ulang.gg=krsm.gg', 'right');
        $this->db->where('pegawai.IdUser', $data->id);
        $this->db->where('krsm.thakademik', $thakademik[0]-
>thakademik);
        $this->db->where('krsm.gg', $thakademik[0]->gg);
        $resultkrsm = $this->db->get('krsm');
        foreach ($resultkrsm->result_object() as $key => $value) {
            $value->status = "Finish";
            array_push($histori, $value);
        }
    }

    $DatasTemKrsm = array(
        'TemKrsm' => array(),
        'Histori' => $histori
    );
    foreach ($temKrsm->result() as $value) {

```

```

$itemTemKrsrm = [
    'Id' => $value->Id,
    'thakademik' => $value->thakademik,
    'gg' => $value->gg,
    'npm' => $value->npm,
    'sms' => $value->sms,
    'dsn_wali' => $value->dsn_wali,
    'ketjur' => $value->ketjur,
    'admakademik' => $value->admakademik,
    'jmsks' => $value->jmsks,
    'tgkrsrm' => $value->tgkrsrm,
    'status' => $value->status,
    'nmmhs' => $value->nmmhs,
    'kelas' => $value->kelas,
    'detailTemKrsrm' => array(),
];
$this->db->where('IdKrsrm', $value->Id);
$itemDetailTem = $this->db->get('tem_detail_krsrm');
array_push($itemTemKrsrm['detailTemKrsrm'], $itemDetailTem-
>result());
array_push($DatasTemKrsrm['TemKrsrm'], $itemTemKrsrm);
}
return $DatasTemKrsrm;
}

```

```

public function UpdateTemKrsrm($item)
{
    if ($item['status'] == "Keuangan") {
        $itemset = "Dosen Wali";
        $Id = $item['Id'];
        $this->db->set('status', $itemset);
        $this->db->where('Id', $Id);
        $hasil = $this->db->update('tem_krsrm');
        return $hasil;
    } else if ($item['status'] == "Dosen Wali") {
        $itemset = "Prodi";
        $Id = $item['Id'];
        $this->db->set('status', $itemset);
        $this->db->where('Id', $Id);
        $hasil = $this->db->update('tem_krsrm');
        return $hasil;
    } else {
        $itemData = (object) $item;
        $this->db->trans_begin();
        $Data_Krsrm = [
            'thakademik' => $itemData->thakademik,

```



```

'gg' => $ItemData->gg,
'npm' => $ItemData->npm,
'sms' => $ItemData->sms,
'dsn_wali' => $ItemData->dsn_wali,
'ketjur' => $ItemData->ketjur,
'admakademik' => $ItemData->admakademik,
'jmsks' => $ItemData->jmsks,
'tgkrsm' => $ItemData->tgkrsm,
];
$numkrs = $this->db->query(
    "SELECT IdKrsm from krsm WHERE thakademik = '$ItemData-
>thakademik' AND gg='$ItemData->gg' AND npm = '$ItemData->npm'"
);
if($numkrs->num_rows()===0){
    $this->db->insert('krsm', $Data_Krsm);
    $IdKrsm = $this->db->insert_id();
    foreach ($ItemData->detailTemKrsm[0] as $key => $value) {
        $kmk = $value['kmk'];
        $numDetailKrsm = $this->db->query(
            "SELECT Id from krsm_detail WHERE thakademik =
'$ItemData->thakademik' AND gg='$ItemData->gg' AND npm = '$ItemData-
>npm' AND kmk = '$kmk'"
        );
        if($numDetailKrsm->num_rows()===0){
            $DetailTemKrsm = array(
                'thakademik' => $ItemData->thakademik,
                'gg' => $value['gg'],
                'npm' => $value['npm'],
                'kmk' => $value['kmk'],
                'nidn' => $value['nidn'],
                'dsnampu' => $value['dsnampu'],
                'nmmk' => $value['nmmk'],
                'bup' => $value['bup'],
                'sks' => $value['sks'],
                'smt' => $value['smt'],
                'kelas' => $value['kelas'],
                'IdKrsm' => $IdKrsm,
            );
            $this->db->insert('krsm_detail', $DetailTemKrsm);
        }
    }
}
}
}

```

```

$numKhsm = $numkrs = $this->db->query(
    "SELECT Id from khsm WHERE thakademik = '$ItemData-
>thakademik' AND gg='$ItemData->gg' AND npm = '$ItemData->npm'"

```

```

);
if($numKhsm->num_rows()===0){
    $DataKhsm = array(
        'thakademik' => $ItemData->thakademik,
        'gg' => $value['gg'],
        'npm' => $value['npm'],
        'dsnwali' => $ItemData->dsn_wali,
        'ketjur' => $ItemData->ketjur,
        'admakademik' => $ItemData->admakademik,
        'jmsks' => $ItemData->jmsks,
    );
    $this->db->insert('khsm', $DataKhsm);
    $IdKhsm = $this->db->insert_id();
    foreach ($ItemData->detailTemKrsm[0] as $key => $value) {
        $kmk = $value['kmk'];
        $numDetailKhsm = $this->db->query(
            "SELECT Id from khsm_detail WHERE thakademik =
'$ItemData->thakademik' AND gg='$ItemData->gg' AND npm = '$ItemData-
>npm' AND kmk = '$kmk'"
        );
        if($numDetailKhsm->num_rows()===0){
            $DetailTemKrsm = array(
                'thakademik' => $ItemData->thakademik,
                'gg' => $value['gg'],
                'npm' => $value['npm'],
                'kmk' => $value['kmk'],
                'IdKhsm' => $IdKhsm
            );
            $this->db->insert('khsm_detail', $DetailTemKrsm);
        }
    }
}
$this->db->where('IdKrsm', $ItemData->Id);
$this->db->delete($this->KrsmDetailTabel);
$this->db->where('Id', $ItemData->Id);
$this->db->delete($this->KrsmTable);
$this->db->set('stdu', 'AKTIF');
$this->db->where('thakademik', $ItemData->thakademik);
$this->db->where('gg', $ItemData->gg);
$this->db->where('npm', $ItemData->npm);
$this->db->where('stdu !=', 'PINDAH/TRANS');
$resultDU = $this->db->update('daftar_ulang');
if($resultDU){
    $this->db->set('statuskul', 'AKTIF');
    $this->db->where('npm', $ItemData->npm);
    $this->db->update('mahasiswa');
}

```

```

        if ($this->db->trans_status() === false) {
            $this->db->trans_rollback();
            return false;
        } else {
            $this->db->trans_commit();
            return true;
        }
    } else {
        $this->db->trans_rollback();
        return false;
    }
}
}

```

```

public function DeleteItemKRSM($item)
{
    $this->db->where('Id', $item);
    $result = $this->db->delete('tem_detail_krsm');
    return $result;
}

```

```

public function Inser_Pengajuan($data, $b)
{
    $this->db->where('npm', $data->npm);
    $this->db->where('thakademik', $data->thakademik);
    $this->db->where('gg', $data->gg);
    $CekKrsn = $this->db->get($this->KrsnTable);
    if ($CekKrsn->num_rows() != 0) {
        $this->db->where('thakademik', $data->thakademik);
        $this->db->where('gg', $data->gg);
        $this->db->where('kmk', $b->kmk);
        $this->db->where('kdps', $data->kdps);
        $Dosenampu = $this->db->get($this->PengampuhTabel);
        $this->db->where('npm', $data->npm);
        $this->db->where('kmk', $b->kmk);
        $CekMatakuliah = $this->db->get('khsm_detail');
        $BUP = "";
        if ($CekMatakuliah->num_rows() == 0) {
            $BUP = "B";
        } else {
            $BUP = "UP";
        }

        $DetailTemKrsn = array(
            'gg' => $data->gg,

```

```

        'npm' => $data->npm,
        'kmk' => $b->kmk,
        'nidn' => $Dosenampu->row('nidn'),
        'dsnampu' => $b->dsn_saji,
        'nmmk' => $b->nmmk,
        'bup' => $BUP,
        'sks' => $b->sks,
        'smt' => $b->smt,
        'kelas' => $b->kelas,
        'IdKrms' => $CekKrms->row('Id'),
    );
    $this->db->insert($this->KrmsDetailTabel, $DetailTemKrms);
    return true;
} else {
    $this->db->select('*');
    $this->db->join('dosen', 'dosen.nidn = dosen_wali.nidn', 'left');
    $this->db->where('npm', $data->npm);
    $GetWali = $this->db->get($this->DosenWaliTabel);
    if ($GetWali->num_rows()) {
        $this->db->where('kdps', $data->kdps);
        $GetJurusan = $this->db->get($this->ProgramStudiTabel);
        if ($GetJurusan->num_rows()) {
            $this->db->where('status', 'Y');
            $baak = $this->db->get($this->BaakTabel);
            if ($baak->num_rows()) {
                $this->db->trans_begin();
                $JumKRS = $this->db->query("SELECT * FROM
daftar_ulang WHERE npm='$data->npm'");
                $Semester = $JumKRS->num_rows();
                $Data_Krms = [
                    'thakademik' => $data->thakademik,
                    'gg' => $data->gg,
                    'npm' => $data->npm,
                    'sms' => $Semester,
                    'dsn_wali' => $GetWali->row('nmdsn'),
                    'ketjur' => $GetJurusan->row('kaprodi'),
                    'admakademik' => $baak->row('adm'),
                    'jmsks' => $data->jmsks,
                    'tgkrms' => date("Y-m-d"),
                ];
                $this->db->insert($this->KrmsTable, $Data_Krms);
                $IdTemKrms = $this->db->insert_id();
                foreach ($b as $key => $value) {
                    $this->db->where('thakademik', $data->thakademik);
                    $this->db->where('gg', $data->gg);
                    $this->db->where('kmk', $value['kmk']);
                }
            }
        }
    }
}

```

```
$this->db->where('kdps', $data->kdps);
$Dosenampu = $this->db->get($this->PengampuhTabel);
$this->db->where('npm', $data->npm);
$this->db->where('kmk', $value['kmk']);
$CekMatakuliah = $this->db->get('khsm_detail');
$BUP = "";
if ($CekMatakuliah->num_rows() == 0) {
    $BUP = "B";
} else {
    $BUP = "UP";
}

$DetaiTemKrsm = array(
    'gg' => $value['gg'],
    'npm' => $data->npm,
    'kmk' => $value['kmk'],
    'nidn' => $Dosenampu->row('nidn'),
    'dsnampu' => $value['dsn_saji'],
    'nmmk' => $value['nmmk'],
    'bup' => $BUP,
    'sks' => $value['sks'],
    'smt' => $value['smt'],
    'kelas' => $value['kelas'],
    'ldKrsm' => $ldTemKrsm,
);
$this->db->insert($this->KrsmDetailTabel,
$DetaiTemKrsm);
}
if ($this->db->trans_status() === false) {
    $this->db->trans_rollback();
    return false;
} else {
    $this->db->trans_commit();
    return true;
}
}
}
} else {
    return false;
}
}
}
```

## 6. KRSM Mahasiswa

### a. Controller

```
<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

use Restserver\Libraries\REST_Controller;

require APPPATH . '/libraries/REST_Controller.php';

class KrsmMahasiswa extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct();
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, PUT, DELETE,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('KrsmMahasiswa_model',
'KrsmMahasiswaModel');
    }
    public function GetAll_get()
    {
        $npm = $this->get('npm');
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $npm = $this->get('npm');
            if (!$npm) {
                $message = [
                    'status' => false,
                    'message' => 'Tidak ada Parameter',
                    'data' => NULL,
                ];
                $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
            }else{
                $result = $this->KrsmMahasiswaModel-
>GetOneByTAAktif($npm);
                if ($result) {
                    $message = [
                        'status' => false,
                        'message' => 'Tidak ada Parameter',
                        'data' => $result,
```

```

];
$this->response($message, REST_Controller::HTTP_OK);
} else {
    $message = [
        'status' => false,
        'message' => 'Tidak ada Data',
        'data' => NULL,
    ];
    $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
    }
    }
    }
    }
}

```

## b. Model

<?php

```

class KrsmMahasiswa_Model extends CI_Model
{
    public function GetOneByTAAktif($npm)
    {
        $this->db->where('status', "AKTIF");
        $queryTA = $this->db->get('tahun_akademik');
        if ($queryTA->num_rows() == 1) {
            $this->db->where('npm', $npm);
            $this->db->where('thakademik', $queryTA->row('thakademik'));
            $this->db->where('gg', $queryTA->row('gg'));
            $resultDetailKrsm = $this->db->get('krsm_detail');
            if ($resultDetailKrsm->num_rows()>0){
                return $resultDetailKrsm->result_array();
            }else{
                return 0;
            }
        }
    }
}

```

## 7. KHSM

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

```

```

use Restserver\Libraries\REST_Controller;

```

```

require APPPATH . '/libraries/REST_Controller.php';

class Khsm extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Khsm_model', 'KhsmModel');
    }

    public function GetKhsm_get()
    {
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        $npm = $this->get('npm');
        if ($is_valid_token['status'] === true) {
            $Output = $this->KhsmModel->AmbilKhs($npm);
            if (!empty($Output && $Output != false)) {
                $message = [
                    'status' => true,
                    'data' => (object) $Output,
                    'message' => "Success",
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            }
        }
    }

    public function AmbillPK_get()
    {
        $npm = $this->get('npm');
        $Output = $this->KhsmModel->AmbillPS($npm);
        if (!empty($Output && $Output != false)) {
            $Datas = array(
                'Data' => []
            );
            for ($i=1; $i <=8 ; $i++) {
                $item = [
                    'Semester' => "",
                    'IPS' => "",
                ];
            }
        }
    }
}

```



```

        'SKS' => "",
        'NSKS' => "",
        'ListMatakuliah' => array()
    ];
    $nilai = 0;
    $sks = 0;
    foreach ($Output['Datas'] as $key => $value) {
        if($value['smt']==$i){
            $nilai += (int)$value['nxsks'];
            $sks += (int)$value['sks'];
            array_push($item['ListMatakuliah'], $value);
        }
    }
    $IPS = $nilai / $sks;
    $item['Semester']= $i;
    $item['IPS']=$IPS;
    $item['SKS']=$sks;
    $item['NSKS']=$nilai;
    array_push($Datas['Data'], $item);

}
$message = [
    'status' => true,
    'data' => $Datas,
    'message' => "Success",
];
$this->response($message, REST_Controller::HTTP_OK);
}
}
public function CreateKHS_post()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $item = json_decode($this->security->xss_clean($this->input->raw_input_stream), true);
        $Output = $this->KhsmModel->insert($item);
        if ($Output) {
            $message = [
                'status' => true,
                'data' => (object) $Output,
                'message' => "Success",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [

```

```

        'status' => false,
        'message' => "Gagal Simpan",
    ];
    $this->response($message, REST_Controller::HTTP_OK);
}
}else{
    $message = [
        'status' => false,
        'message' => "Session Anda Habis"
    ];
    $this->response($message, Rest_Controller::HTTP_OK);
}
}
public function GetlistKHS_get()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {
        $Output = $this->KhsModel->getNilai($is_valid_token['data']);
        if (!empty($Output)) {
            $message = [
                'status' => true,
                'data' => (object) $Output,
                'message' => "Success",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
    $message = [
        'status' => false,
        'message' => "Kosong",
    ];
    $this->response($message, REST_Controller::HTTP_OK);
}
}
}else{
    $message = [
        'status' => false,
        'message' => "Session Anda Habis"
    ];
    $this->response($message, Rest_Controller::HTTP_OK);
}
}
}
public function PutDetaiKHS_put()
{
    $this->load->library('Authorization_Token');
    $is_valid_token = $this->authorization_token->validateToken();
    if ($is_valid_token['status'] === true) {

```

```

        $item = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        $Output = $this->KhsmModel->putNilai($item);
        if ($Output) {
            $message = [
                'status' => true,
                'message' => "Success",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [
                'status' => false,
                'message' => "Gagal",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }else{
        $message = [
            'status' => false,
            'message' => "Session Anda Habis"
        ];
        $this->response($message, Rest_Controller::HTTP_OK);
    }
}
}
}

```

## b. Model

<?php

```

class Khsm_Model extends CI_Model
{
    public function AmbilKhs($npm)
    {
        $this->db->where('npm', $npm);
        $result = $this->db->get('mahasiswa');
        $itemMahasiswa = $result->result();
        $kurikulum = $result->row('kurikulum');
        $this->db->where('status_mk', 'OK');
        $this->db->where('kdps', $result->row('kdps'));
        $this->db->group_start();
        $this->db->where('kurikulum', $kurikulum);
        $this->db->or_where('kurikulum', 'ALL');
        $this->db->group_end();
        $this->db->order_by('matakuliah.smt', 'ASC');
        $result = $this->db->get('matakuliah');
        $matakuliah = $result->result();
    }
}

```

```

$this->db->where('status_mk', 'NO');
$this->db->where('kdps', $result->row('kdps'));
$this->db->group_start();
$this->db->where('kurikulum', $kurikulum);
$this->db->or_where('kurikulum', 'ALL');
$this->db->group_end();
$this->db->order_by('matakuliah.smt', 'ASC');
$result = $this->db->get('matakuliah');
$matakuliahNo = $result->result();
$this->db->where('npm', $npm);
$result = $this->db->get('transkip');
$itemKhsm = $result->result();
$KhsmData = array(
    'Datas' => array(),
    'IPK' => "",
    'SKS' => "",
);
$TotalSks = 0;
$Totalnxsk = 0;
foreach ($matakuliah as $valueMatakuliah) {
    $item = array(
        'nmmk' => $valueMatakuliah->nmmk,
        'smt' => $valueMatakuliah->smt,
        'mk_konversi' => $valueMatakuliah->mk_konversi,
        'sks' => $valueMatakuliah->sks,
        'npm' => "-",
        'kmk' => $valueMatakuliah->kmk,
        'nhuruf' => "-",
        'nxsk' => "-",
        'ngBinding' => "",
        'ket' => "-",
    );
    foreach ($itemKhsm as $key => $valueKhsm) {
        $nilai = 0;

        if ($valueMatakuliah->kmk == $valueKhsm->kmk) {
            if ((int) $valueKhsm->nxsk > $nilai) {
                $warna = "";
                if ($valueKhsm->nilai == "A" || $valueKhsm->nilai == "B+" ||
                    $valueKhsm->nilai == "B" || $valueKhsm->nilai == "C+") {
                    $warna = "info";
                } else if ($valueKhsm->nilai == "c") {
                    $warna = "danger";
                } else {
                    $warna = "warning";
                }
            }
        }
    }
}

```

```

$item['npm'] = $valueKhsm->npm;
$item['kmk'] = $valueKhsm->kmk;
$item['nhuruf'] = $valueKhsm->nilai;
$item['nxsks'] = $valueKhsm->nxsks;
$item['ngBinding'] = $warna;
$item['ket'] = $valueKhsm->ket;
if ($valueKhsm->ket !== "TL") {
    $nilai = (int) $valueKhsm->nxsks;
    $TotalSks += (int) $valueMatakuliah->sks;
}
}else{
    $warna = "warning";
    $item['npm'] = $valueKhsm->npm;
    $item['kmk'] = $valueKhsm->kmk;
    $item['nhuruf'] = $valueKhsm->nilai;
    $item['nxsks'] = $valueKhsm->nxsks;
    $item['ngBinding'] = $warna;
    $item['ket'] = $valueKhsm->ket;
}
} else {
    foreach ($matakuliahNo as $valueNo) {
        if ($valueMatakuliah->kmk == $valueNo->mk_konversi) {
            $nilai = 0;
            if ($valueKhsm->kmk == $valueNo->kmk) {
                if ((int) $valueKhsm->nxsks > $nilai) {
                    $warna = "";
                    if ($valueKhsm->nilai == "A" || $valueKhsm->nilai ==
"B+" || $valueKhsm->nilai == "B" || $valueKhsm->nilai == "C+") {
                        $warna = "info";
                    } else if ($valueKhsm->nilai == "c") {
                        $warna = "danger";
                    } else {
                        $warna = "warning";
                    }
                }

                $item['npm'] = $valueKhsm->npm;
                $item['kmk'] = $valueKhsm->kmk;
                $item['nhuruf'] = $valueKhsm->nilai;
                $item['nxsks'] = $valueKhsm->nxsks;
                $item['ngBinding'] = $warna;
                $item['ket'] = $valueKhsm->ket;
                if ($valueKhsm->ket !== "TL") {
                    $nilai = (int) $valueKhsm->nxsks;
                    $TotalSks += (int) $valueMatakuliah->sks;
                }
            }
        }
    }
}

```

```

    }
    }
    }
    }
    $TotalInxsks += (int) $nilai;
}
array_push($KhsmData['Datas'], $item);

}
$IPK = 0;
if ($TotalInxsks == 0) {
    $IPK = 0;
} else {
    $IPK = round($TotalInxsks / $TotalSks, 2);
}

$KhsmData['IPK'] = $IPK;
$KhsmData['SKS'] = $TotalSks;
return $KhsmData;
}

public function AmbillIPS($npm)
{
    $this->db->where('npm', $npm);
    $result = $this->db->get('mahasiswa');
    $itemMahasiswa = $result->result();
    $kurikulum = $result->row('kurikulum');
    $this->db->where('status_mk', 'OK');
    $this->db->where('kdps', $result->row('kdps'));
    $this->db->group_start();
    $this->db->where('kurikulum', $kurikulum);
    $this->db->or_where('kurikulum', 'ALL');
    $this->db->group_end();
    $this->db->order_by('matakuliah.smt', 'ASC');
    $result = $this->db->get('matakuliah');
    $matakuliah = $result->result();
    $this->db->where('status_mk', 'NO');
    $this->db->where('kdps', $result->row('kdps'));
    $this->db->group_start();
    $this->db->where('kurikulum', $kurikulum);
    $this->db->or_where('kurikulum', 'ALL');
    $this->db->group_end();
    $this->db->order_by('matakuliah.smt', 'ASC');
    $result = $this->db->get('matakuliah');
    $matakuliahNo = $result->result();
    $this->db->where('npm', $npm);

```

```

$result = $this->db->get('transkip');
$itemKhsm = $result->result();
$KhsmData = array(
    'Datas' => array(),
    'IPK' => "",
    'SKS' => "",
);
$TotalSks = 0;
$Totalnxsk = 0;
foreach ($matakuliah as $valueMatakuliah) {
    $item = array(
        'nmmk' => $valueMatakuliah->nmmk,
        'smt' => $valueMatakuliah->smt,
        'mk_konversi' => $valueMatakuliah->mk_konversi,
        'sks' => $valueMatakuliah->sks,
        'npm' => "-",
        'kmk' => $valueMatakuliah->kmk,
        'nhuruf' => "-",
        'nxsk' => "-",
        'ngBinding' => "",
        'ket' => "-",
    );
    foreach ($itemKhsm as $key => $valueKhsm) {
        $nilai = 0;

        if ($valueMatakuliah->kmk == $valueKhsm->kmk) {
            if ((int) $valueKhsm->nxsk > $nilai) {
                $warna = "";
                if ($valueKhsm->nilai == "A" || $valueKhsm->nilai == "B") {
                    $warna = "info";
                } else if ($valueKhsm->nilai == "C") {
                    $warna = "danger";
                } else {
                    $warna = "warning";
                }
            }

            $item['npm'] = $valueKhsm->npm;
            $item['kmk'] = $valueKhsm->kmk;
            $item['nhuruf'] = $valueKhsm->nilai;
            $item['nxsk'] = $valueKhsm->nxsk;
            $item['ngBinding'] = $warna;
            $item['ket'] = $valueKhsm->ket;
            if ($valueKhsm->ket == "L") {
                $nilai = (int) $valueKhsm->nxsk;
                $TotalSks += (int) $valueMatakuliah->sks;
            }
        }
    }
}

```

```

    }
  } else {
    foreach ($matakuliahNo as $valueNo) {
      if ($valueMatakuliah->kmk == $valueNo->mk_konversi) {
        $nilai = 0;
        if ($valueKhsm->kmk == $valueNo->kmk) {
          if ((int) $valueKhsm->nxsks > $nilai) {
            $warna = "";
            if ($valueKhsm->nilai == "A" || $valueKhsm->nilai ==
"B") {

              $warna = "info";
            } else if ($valueKhsm->nilai == "c") {
              $warna = "danger";
            } else {
              $warna = "warning";
            }

            $item['npm'] = $valueKhsm->npm;
            $item['kmk'] = $valueKhsm->kmk;
            $item['nhuruf'] = $valueKhsm->nilai;
            $item['nxsks'] = $valueKhsm->nxsks;
            $item['ngBinding'] = $warna;
            $item['ket'] = $valueKhsm->ket;
            if ($valueKhsm->ket == "L") {
              $nilai = (int) $valueKhsm->nxsks;
              $TotalSks += (int) $valueMatakuliah->sks;
            }
          }
        }
      }
    }
  }
  $Totalnxsks += (int) $nilai;
}
array_push($KhsmData['Datas'], $item);

}
$IPK = 0;
if ($Totalnxsks == 0) {
  $IPK = 0;
} else {
  $IPK = round($Totalnxsks / $TotalSks, 2);
}

$KhsmData['IPK'] = $IPK;
$KhsmData['SKS'] = $TotalSks;

```



```

        return $KhsmData;
    }
    public function insert($data)
    {
        $this->db->trans_begin();
        $resultBA = $this->db->insert('ban_matakuliah', $data->ba);
        $resultkhs = $this->db->insert('khsm', $data->khsm);
        $idkhs = $this->db->insert_id();
        foreach ($data['detailkhsm'] as $key => $value) {
            $detail = [
                'idPengampu' => $value['idPengampu'],
                'idKrs' => $idkhs

            ];
            $this->db->insert("khsm_detail", $value);
        }

        if ($this->db->trans_status() === false) {
            $this->db->trans_rollback();
            return false;
        }else{
            $this->db->trans_commit();
            return true;
        }
    }
    public function getNilai($item)
    {
        $id = $item->id;
        $resultMatakuliah = $this->db->query("
            SELECT
                `jadwal_kuliah`.*,
                `matakuliah`.`smt`,
                `matakuliah`.`kurikulum`,
                `program_studi`.`nmps`,
                `dosen`.`nmdsn`,
                `dosen_pengampu`.`nidn`
            FROM
                `jadwal_kuliah`
            LEFT JOIN `tahun_akademik` ON `jadwal_kuliah`.`thakademik` =
                `tahun_akademik`.`thakademik` AND `jadwal_kuliah`.`gg` =
                `tahun_akademik`.`gg`
            LEFT JOIN `matakuliah` ON `matakuliah`.`kmk` =
                `jadwal_kuliah`.`kmk`
            AND `matakuliah`.`kdps` = `jadwal_kuliah`.`kdps`
            LEFT JOIN `program_studi` ON `jadwal_kuliah`.`kdps` =
                `program_studi`.`kdps`

```

```

        LEFT JOIN `dosen_pengampu` ON `jadwal_kuliah`.`kmk` =
        `dosen_pengampu`.`kmk`
        RIGHT JOIN `dosen` ON `dosen`.`nidn` =
        `dosen_pengampu`.`nidn`
        RIGHT JOIN `pegawai` ON `pegawai`.`idpegawai` =
        `dosen`.`idpegawai`
    WHERE
        `tahun_akademik`.`status` = 'AKTIF' AND
        `dosen_pengampu`.`mengajar` = 'Y' AND
        `pegawai`.`idUser` = '$id'
    ORDER BY
        `matakuliah`.`nmmk`
");
$a = $resultMatakuliah->result_object();
foreach ($a as $key => $value) {
    $kmk = $value->kmk;
    $resultMahasiswa = $this->db->query("
        SELECT
            `mahasiswa`.`npm`,
            `mahasiswa`.`jenjang`,
            `mahasiswa`.`nmmhs`,
            `khsmdetail`.`thakademik`,
            `khsmdetail`.`gg`,
            `khsmdetail`.`nhuruf`,
            `khsmdetail`.`nxsk`,
            `khsmdetail`.`ket`,
            `krsmdetail`.`sk`,
            `matakuliah`.`nmmk`,
            `khsmdetail`.`kmk`,
            `khsmdetail`.`nilai`,
            `krsmdetail`.`kelas`
        FROM
            `khsmdetail`
        LEFT JOIN `mahasiswa` ON `mahasiswa`.`npm` =
        `khsmdetail`.`npm`
        LEFT JOIN `tahun_akademik` ON
        `tahun_akademik`.`thakademik` =
        `khsmdetail`.`thakademik` AND `tahun_akademik`.`gg` =
        `khsmdetail`.`gg`
        LEFT JOIN `krsmdetail` ON `krsmdetail`.`thakademik` =
        `khsmdetail`.`thakademik` AND `krsmdetail`.`gg` =
        `khsmdetail`.`gg` AND
        `krsmdetail`.`npm` = `khsmdetail`.`npm` AND
        `krsmdetail`.`kmk` =
        `khsmdetail`.`kmk`
    ");
}

```

```

        LEFT JOIN `matakuliah` ON `khsmdetail`.`kmk` =
`matakuliah`.`kmk`
        WHERE
            `khsmdetail`.`kmk` = '$kmk' AND
            `tahun_akademik`.`status` = 'AKTIF' AND
            `krsm_detail`.`kelas` = '$value->kelas' AND
            matakuliah.status_mk = 'OK'

    );
    $b = $resultMahasiswa->result_object();
    foreach ($b as $key3 => $value3) {
        $value3->nilai = floatval($value3->nilai);
    }
    $value->Mahasiswa = $b;
}
return $a;

}
public function putNilai($item)
{
    $this->db->where("thakademik", $item['thakademik']);
    $this->db->where("gg", $item['gg']);
    $this->db->where("nidn", $item['nidn']);
    $this->db->where("kdps", $item['kdps']);
    $this->db->where("kmk", $item['kmk']);
    $this->db->where("kelas", $item['kelas']);
    $result = $this->db->get("ban_matakuliah");
    $num = $result->num_rows();

    $this->db->trans_start();
    if($num == 0){
        $tg = date("Y-m-d");
        $data = [
            "thakademik" => $item['thakademik'],
            "gg" => $item['gg'],
            "nidn" => $item['nidn'],
            "kdps" => $item['kdps'],
            "kmk" => $item['kmk'],
            "nmdsn" => $item['nmdsn'],
            "nmpps" => $item['nmpps'],
            "nmmk" => $item['nmmk'],
            "kelas" => $item['kelas'],
            "tgbn" => $tg
        ];
        $this->db->insert("ban_matakuliah", $data);
    }
}

```

```

foreach ($item['Mahasiswa'] as $key => $value) {
    $this->db->set("nhuruf", $value['nhuruf']);
    $this->db->set("nxsk", $value['nxsk']);
    $this->db->set("nilai", $value['nilai']);
    $this->db->set("ket", $value['ket']);
    $this->db->where("thakademik", $value['thakademik']);
    $this->db->where("gg", $value['gg']);
    $this->db->where("npm", $value['npm']);
    $this->db->where("kmk", $value['kmk']);
    $this->db->update("khsm_detail");
    $this->db->where("npm", $value['npm']);
    $this->db->where("kdps", $item['kdps']);
    $this->db->where("kmk", $value['kmk']);
    $result = $this->db->get('transkip');
    if($result->num_rows()>0){
        $DataTranskip = $result->result_object();
        if($value['nxsk'] > $DataTranskip[0]->nxsk){
            $this->db->set("nxsk", $value['nxsk']);
            $this->db->set("nilai", $value['nhuruf']);
            $this->db->set("ket", $value['ket']);
            $this->db->where("npm", $value['npm']);
            $this->db->where("kdps", $item['kdps']);
            $this->db->where("kmk", $value['kmk']);
            $this->db->update("transkip");
        }
    }
}
if ($this->db->trans_status() === false) {
    $this->db->trans_rollback();
    return false;
}else{
    $this->db->trans_complete();
    return true;
}

}
}

```

## 8. Matakuliah

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

```

```

use Restserver\Libraries\REST_Controller;

```

```

require APPPATH . '/libraries/REST_Controller.php';

class Matakuliah extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        $this->load->model('Matakuliah_model', 'MatakuliahModel');
    }
    public function GetMatakuliah_get()
    {
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: GET");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");

        $Output = $this->MatakuliahModel->AmbilMatakuliah();
        $message = [
            'status' => true,
            'data' => $Output
        ];
        $this->response($message, REST_Controller::HTTP_OK);
    }
    public function GetKrsM_get()
    {
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: GET");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $Output = $this->MatakuliahModel-
>ambilKrsM($is_valid_token['data']);
            $message = [
                'status' => true,
                'data' => $Output
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }else{
            $message = [
                'status' => true,
                'data' => [],
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

```

```

        'message' => "Session Anda telah kadaluarsa"
    ];
    $this->response($message, REST_Controller::HTTP_OK);
}

}

}

```

## b. Model

```
<?php
```

```

class Matakuliah_Model extends CI_Model
{
    public function AmbilMatakuliah()
    {
        $result = $this->db->query("
            SELECT
                `program_studi`.`kdps`,
                `program_studi`.`nmps`,
                `program_studi`.`jenjang`,
                `matakuliah`.*
            FROM
                `program_studi`
            LEFT JOIN `matakuliah` ON `matakuliah`.`kdps` =
`program_studi`.`kdps`
            WHERE
                program_studi.status = 'true' and
                matakuliah.kurikulum in(2011,2018,2019)
        ");
        return $result->result_array();
    }

    public function ambilkrsm($data)
    {
        $result = $this->db->query("
            SELECT
                `krsm_detail`.*,
                `dosen_pengampu`.`kdps`
            FROM
                `krsm_detail`
            LEFT JOIN `tahun_akademik` ON
`tahun_akademik`.`thakademik` =
                `krsm_detail`.`thakademik` AND `tahun_akademik`.`gg` =
`krsm_detail`.`gg`
            LEFT JOIN `dosen_pengampu` ON `dosen_pengampu`.`kmk` =
`krsm_detail`.`kmk`
        ");
        return $result->result_array();
    }
}

```

```

        LEFT JOIN `jadwal_kuliah` ON `jadwal_kuliah`.`idpengampu` =
        `dosen_pengampu`.`idpengampu` AND `jadwal_kuliah`.`kelas` =
        `krsm_detail`.`kelas`
        LEFT JOIN `mahasiswa` ON `krsm_detail`.`npm` =
        `mahasiswa`.`npm`
    WHERE
        `mahasiswa`.`idUser` = '$data->id' AND
        `tahun_akademik`.`status` = 'AKTIF' AND
        `dosen_pengampu`.`mengajar` = 'Y'
    ");
    return $result->result_array();
}
}

```

## 9. Mahasiswa Wali

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

use Restserver\Libraries\REST_Controller;

require APPPATH . '/libraries/REST_Controller.php';

class MahasiswaWali extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE,
PUT");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('MahasiswaWali_model', 'MahasiswaWaliModel');
    }
    public function GetMahasiswaWali_get()
    {
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $Output = $this->MahasiswaWaliModel-
>Select($is_valid_token['data']->id);
            if (!empty($Output)) {
                $message = [
                    'status' => true,
                    'data' => $Output,

```

```

];
$this->response($message, REST_Controller::HTTP_OK);
    }
}
}
}
}

```

## b. Model

```
<?php
```

```

class MahasiswaWali_model extends CI_Model
{
    public function Select($idUser)
    {
        $result = $this->db->query(
            "SELECT
            `mahasiswa`.`npm`,
            `mahasiswa`.`kelas`,
            `mahasiswa`.`nmmhs`,
            `mahasiswa`.`almt`,
            `mahasiswa`.`notlp`,
            `mahasiswa`.`statuskul`,
            `mahasiswa`.`jk`,
            `program_studi`.`kdps`,
            `program_studi`.`nmps`,
            `program_studi`.`jenjang`,
            (SELECT
            SUM(CASE
            WHEN transkip.ket= 'L' THEN 1*transkip.nxsks
            ELSE
            0*transkip.nxsks
            END)/SUM(CASE
            WHEN transkip.ket = 'L' THEN 1*transkip.sks
            ELSE
            0*transkip.sks
            END) FROM transkip WHERE transkip.npm=`mahasiswa`.`npm`)
            AS IPK ,

            (SELECT
            SUM(CASE
            WHEN transkip.ket = 'L' THEN 1*transkip.sks
            ELSE
            0*transkip.sks
            END)FROM transkip WHERE transkip.npm=`mahasiswa`.`npm`)
            AS SKSLulus,

```



```

        (SELECT COUNT(daftar_ulang.thakademik) FROM daftar_ulang
where daftar_ulang.npm = `mahasiswa`.`npm`) AS Semester
FROM
    `dosen_wali`
    LEFT JOIN `mahasiswa` ON `dosen_wali`.`npm` =
`mahasiswa`.`npm`
    RIGHT JOIN `program_studi` ON `mahasiswa`.`kdps` =
`program_studi`.`kdps`
    RIGHT JOIN `dosen` ON `dosen`.`iddosen` =
`dosen_wali`.`iddosen`
    RIGHT JOIN `pegawai` ON `pegawai`.`idpegawai` =
`dosen`.`idpegawai`
    WHERE mahasiswa.statuskul IN('AKTIF','TIDAK AKTIF','CUTI') AND
pegawai.IdUser='$IdUser'
    );
    return $result->result_object();
}
}

```

## 10. Pengampuh/Pengajar

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

use Restserver\Libraries\REST_Controller;

require APPPATH . '/libraries/REST_Controller.php';

class Pengampuh extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Pengampuh_model', 'PengampuhModel');
    }

    public function GetPengampuh_get()
    {
        // $_POST = $this->security->xss_clean($_POST);
        $this->load->library('Authorization_Token');
    }
}

```

```

        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $Id = $_GET;
            $result = $this->PengampuhModel->get($Id);
            $message = [
                'status' => true,
                'data' => $result,
                'message' => "Success",
            ];
            $this->response($message, REST_Controller::HTTP_OK);
        }
    }
}

```

## b. Model

<?php

```

class Pengampuh_Model extends CI_Model
{
    public function get($Id)
    {
        $id = $Id['idpengampu'];
        $result = $this->db->query("
            SELECT
                `dosen`.*
            FROM
                `dosen_pengampu`
            LEFT JOIN `dosen` ON `dosen`.`nidn` = `dosen_pengampu`.`nidn`
            WHERE
                `dosen_pengampu`.idpengampu = '$id' AND
                `dosen_pengampu`.`mengajar`='Y'
        ");
        return $result->result_array();
    }
}

```

## 11. Profile

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

```

```

use Restserver\Libraries\REST_Controller;

```

```

require APPPATH . '/libraries/REST_Controller.php';

```

```

class Profile extends \Restserver\Libraries\REST_Controller

```

```

{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Profile_model', 'ProfileModel');
    }
    public function GetProfile_get()
    {
        $role = $this->get('role');
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $Output = $this->ProfileModel->getProfilee($is_valid_token['data'],
$role);
            if ($Output > 0 && !empty($Output)) {
                $message = [
                    'status' => true,
                    'data' => json_encode($Output),
                    'message' => "Success!",
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            }else{
                $message = [
                    'status' => false,
                    'data' => [],
                    'message' => "Data Kosong",
                ];
                $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
            }
        }
    }
    public function UpdateProfile_put()
    {
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        $_PUT = json_decode($this->security->xss_clean($this->input-
>raw_input_stream), true);
        if ($is_valid_token['status'] === true) {

```

```

$Output = $this->ProfileModel->UpdateProfile($_PUT);
if (!empty($Output && $Output != false)) {
    $message = [
        'status' => true,
        'message' => "Berhasil melakukan Approved"
    ];
    $this->response($message, REST_Controller::HTTP_OK);
}else{
    $message = [
        'status' => false,
        'message' => "Proses Gagal",
    ];
    $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
}
}
}
}

```

#### **b. Model**

<?php

```

class Profile_Model extends CI_Model
{
    public function getProfilee($data, $role)
    {
        if ($role == "Mahasiswa") {
            $this->db->where('IdUser', $data->id);
            $result = $this->db->get('mahasiswa');
            return $result->result_array();
        } else {
            $this->db->where('IdUser', $data->id);
            $result = $this->db->get('pegawai');
            return $result->result_array();
        }
    }
    public function UpdateProfile($data)
    {
        if ($data['Role'] == "Pegawai") {
            $this->db->set('Nama', $data['Nama']);
            $this->db->set('Alamat', $data['Alamat']);
            $this->db->set('Kontak', $data['Kontak']);
            $this->db->set('TempatLahir', $data['TempatLahir']);
            $this->db->set('TanggalLahir', $data['TanggalLahir']);
            $this->db->set('NoSK', $data['NoSK']);
        }
    }
}

```

```

        $this->db->set('NIK', $data['NIK']);
        $this->db->set('Nama', $data['Nama']);
        $this->db->where('idpegawai', $data['idpegawai']);
        $result = $this->db->update('pegawai');
        return $result;
    }else{
        $this->db->set('nmmhs', $data['nmmhs']);
        $this->db->set('tmlhr', $data['tmlhr']);
        $this->db->set('tglhr', $data['tglhr']);
        $this->db->set('agama', $data['agama']);
        $this->db->set('kewarga', $data['kewarga']);
        $this->db->set('pendidikan', $data['pendidikan']);
        $this->db->set('nmsmu', $data['nmsmu']);
        $this->db->set('jursmu', $data['jursmu']);
        $this->db->set('kotasmu', $data['kotasmu']);
        $this->db->set('provsmu', $data['provsmu']);
        $this->db->set('pekerjaan', $data['pekerjaan']);
        $this->db->set('almt', $data['almt']);
        $this->db->set('notlp', $data['notlp']);
        $this->db->set('status', $data['status']);
        $this->db->set('jmsaudara', $data['jmsaudara']);
        $this->db->set('nmayah', $data['nmayah']);
        $this->db->set('nmibu', $data['nmibu']);
        $this->db->set('sumbiaya', $data['sumbiaya']);
        $this->db->where('npm', $data['npm']);
        $result = $this->db->update('mahasiswa');
        return $result;
    }
}
}

```

## 12. ApprovedKRSM

### a. Controller

```

<?php defined('BASEPATH') or exit('No direct script access allowed');
date_default_timezone_set('Asia/Tokyo');

```

```

use Restserver\Libraries\REST_Controller;

```

```

require APPPATH . '/libraries/REST_Controller.php';

```

```

class ApprovedKrsM extends \Restserver\Libraries\REST_Controller
{
    public function __construct($config = 'rest')
    {
        parent::__construct($config);
    }
}

```

```

        header("Access-Control-Allow-Origin: *");
        header("Content-Type: application/json; charset=UTF-8");
        header("Access-Control-Allow-Methods: POST, GET, DELETE, PUT,
OPTIONS");
        header("Access-Control-Allow-Headers: Content-Type, Access-
Control-Allow-Headers, Authorization, X-Requested-With");
        $this->load->model('Approved_model', 'ApprovedModel');
    }

    public function ambilHistori_get()
    {
        $role = $this->get('role');
        $this->load->library('Authorization_Token');
        $is_valid_token = $this->authorization_token->validateToken();
        if ($is_valid_token['status'] === true) {
            $Output = $this->ApprovedModel-
>GetHistory($is_valid_token['data'], $role);
            if (!empty($Output)) {
                $message = [
                    'status' => true,
                    'data' => $Output,
                    'message' => "Success!",
                ];
                $this->response($message, REST_Controller::HTTP_OK);
            }else{
                $message = [
                    'status' => false,
                    'data' => [],
                    'message' => "Kosong",
                ];
                $this->response($message,
REST_Controller::HTTP_NOT_FOUND);
            }
        }
    }
}

```

## **b. Model**

<?php

```

class Approved_model extends CI_Model
{
    public function GetHistory($data, $Role)
    {
        if ($Role == "Keuangan") {

```

```
$resultTahunAktif = $this->db->query("SELECT * FROM
tahun_akademik WHERE status='AKTIF'");
```

```
if ($resultTahunAktif->num_rows()) {
    $DataTahunAktif = $resultTahunAktif->row();
    $resultTem = $this->db->query("
    SELECT
        `tem_krsm`.*,
        `mahasiswa`.`kdps`,
        `mahasiswa`.`jenjang`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`
    FROM
        `tem_krsm`
    LEFT JOIN `mahasiswa` ON `mahasiswa`.`npm` =
`tem_krsm`.`npm`
    WHERE thakademik = '$DataTahunAktif->thakademik' AND gg
= '$DataTahunAktif->gg'
    );
    $resultKrsm = $this->db->query("
    SELECT
        `krsm`.*,
        `mahasiswa`.`kdps`,
        `mahasiswa`.`jenjang`,
        `mahasiswa`.`nmmhs`,
        `mahasiswa`.`kelas`
    FROM
        `krsm`
    LEFT JOIN `mahasiswa` ON `mahasiswa`.`npm` =
`krsm`.`npm`
    WHERE thakademik = '$DataTahunAktif->thakademik' AND gg
= '$DataTahunAktif->gg'
    );
    $Message = [
        'Data' => array(),
    ];
```

```
if ($resultTem->num_rows()) {
    foreach ($resultTem->result_array() as $key => $value) {
        if ($value['status'] == 'Keuangan') {
            $Datas = [
                'thakademik' => $value['thakademik'],
                'gg' => $value['gg'],
                'npm' => $value['npm'],
                'sms' => $value['sms'],
                'dsn_wali' => $value['dsn_wali'],
```





```

        data: $scope.DataInput
    }).then(function (response) {
        if (response.data.status)
            var Tampung = response.data;
        var j = Tampung.data;
        var Datarole = JSON.stringify(j.RoleUser.Role);
        $window.sessionStorage.setItem("Token", j.Token);
        $window.sessionStorage.setItem("Username", j.Username);
        $window.sessionStorage.setItem("Email", j.Email);
        $window.sessionStorage.setItem("NamaUser", j.NamaUser);
        $window.sessionStorage.setItem("Role", Datarole);
        window.location.href = "home.html";
    }, function (error) {
        console.log(error);
        alert(error.data.message);
    })
}
})

```

## b. View

```

<!DOCTYPE html>
<html ng-app="Apps">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>SISTEM AKADEMIK STIMIK1011</title>
    <!-- Tell the browser to be responsive to screen width -->
    <meta content="width=device-width, initial-scale=1, maximum-scale=1,
user-scalable=no" name="viewport">
    <link rel="icon" href="assets/img/Logo.png">
    <!-- Bootstrap 3.3.7 -->
    <link                                rel="stylesheet"
href="assets/bower_components/bootstrap/dist/css/bootstrap.min.css">
    <!-- Font Awesome -->
    <link            rel="stylesheet"            href="assets/bower_components/font-
awesome/css/font-awesome.min.css">
    <!-- Ionicons -->
    <link                                rel="stylesheet"
href="assets/bower_components/Ionicons/css/ionicons.min.css">
    <!-- Theme style -->
    <link rel="stylesheet" href="assets/dist/css/AdminLTE.min.css">
    <link rel="stylesheet" href="assets/dist/css/main.css">
    <!--
                                <link                                rel="stylesheet"
href="node_modules/sweetalert2/dist/sweetalert2.css"> -->

```

```
<link rel="stylesheet" href="assets/node_modules/angular-datatables/dist/css/angular-datatables.css">
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/angular-bootstrap-toggle@0.4.0/dist/angular-bootstrap-toggle.min.css">
```

```
<!-- iCheck -->
<link rel="stylesheet" href="plugins/iCheck/square/blue.css">
<script
src="assets/bower_components/jquery/dist/jquery.min.js"></script>
<script src="assets/bower_components/angular/angular.min.js"></script>
<script src="assets/bower_components/angular/angular-
animate.min.js"></script>
<script src="assets/bower_components/angular/angular-
route.min.js"></script>
<script src="assets/bower_components/angular/angular-
sanitize.min.js"></script>
<script src="assets/bower_components/angular/angular-
resource.js"></script>
<script
src="assets/bower_components/angular/angularjs/release/angular-ui-
router.js"></script>
<link rel='stylesheet' href='assets/bower_components/angular/loading-
bar.css' type='text/css' media='all' />
<link rel="stylesheet"
href="assets/bower_components/sweetalert/dist/sweetalert.css">
<script type='text/javascript'
src='assets/bower_components/angular/loading-bar.js'></script>
```

```
<!-- Select2 -->
<!-- <script src="assets/bower_components/angular-select2/dist/angular-
select2.min.js"></script> -->
<script src="assets/bower_components/angular-ui-
select2/src/select2.js"></script>
```

```
<link rel="stylesheet" href="assets/node_modules/angular-
datatables/dist/css/angular-datatables.css">
<script
src="assets/node_modules/datatables.net/js/jquery.dataTables.js"></script>
>
<script src="assets/node_modules/angular-datatables/dist/angular-
datatables.js"></script>
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-
datepicker/1.9.0/css/bootstrap-datepicker.min.css">
<script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-
datepicker/1.9.0/js/bootstrap-datepicker.min.js"></script>
```

<!-- Controller -->

```
<script src="assets/ng/Apps.js"></script>
<script src="assets/ng/Controller.js"></script>
<script src="assets/ng/Controller/Pegawai.js"></script>
<script src="assets/ng/Controller/Jabatan.js"></script>
<script src="assets/ng/Controller/Mahasiswa.js"></script>
<script src="assets/ng/Controller/Jadwal.js"></script>
<script src="assets/ng/Controller/Beranda.js"></script>
<script src="assets/ng/Controller/BaseProfile.js"></script>
<script src="assets/ng/Controller/Prodi.js"></script>
<script src="assets/ng/Controller/TahunAkademik.js"></script>
<script src="assets/ng/Controller/GradeNilai.js"></script>
<script src="assets/ng/Controller/Kurikulum.js"></script>
<script src="assets/ng/Controller/KemajuanStudi.js"></script>
<script src="assets/ng/Controller/ApprovedKRS.js"></script>
<script src="assets/ng/Controller/NilaiMahasiswa.js"></script>
<script src="assets/ng/Controller/ConversiKHS.js"></script>
<script src="assets/ng/Controller/Pengumuman.js"></script>
<script src="assets/ng/Controller/BidikMisi.js"></script>
<script src="assets/ng/Controller/PengajuanKRS.js"></script>
<script src="assets/ng/Controller/Profile.js"></script>
<script src="assets/ng/Controller/Histori.js"></script>
<script src="assets/ng/Controller/Penelitian.js"></script>
<script src="assets/ng/Controller/MahasiswaWali.js"></script>
<script src="assets/ng/Controller/BeritaAcara.js"></script>
<script src="assets/ng/Controller/EvaluasiPembelajaran.js"></script>
<script src="assets/ng/Controller/UserAkses.js"></script>
```

<!-- Directive -->

```
<script src="assets/ng/Directives/ApprovedDirectives.js"></script>
<script src="assets/ng/Directives/PengajuanKRSDirective.js"></script>
<script src="assets/ng/Directives/JadwalDirectives.js"></script>
<script src="assets/ng/Directives/KemajuanStudiDirective.js"></script>
<script src="assets/ng/Directives/UserDirectives.js"></script>
<script src="assets/ng/Directives/BaseProfileDirectives.js"></script>
<script src="assets/ng/Directives/MahasiswaDirectives.js"></script>
<script src="assets/ng/Directives/PenelitianDirectives.js"></script>
<script src="assets/ng/Directives/MahasiswaWaliDirective.js"></script>
<script src="assets/ng/Directives/BeritaAcaraDirective.js"></script>
<script src="assets/ng/Directives/MatakuliahDirectives.js"></script>
<script src="assets/ng/Directives/KhsmDirectives.js"></script>
<script src="assets/ng/Directives/GradeNilaiDirectives.js"></script>
```

```

<script
src="assets/ng/Directives/EvaluasiPembelajaranDirectives.js"></script>
<script src="assets/ng/Directives/TahunAkademikDirectives.js"></script>
<script src="assets/ng/Directives/BidikMisiDirectives.js"></script>
<script src="assets/ng/Directives/PengumumanDirectives.js"></script>

<!-- Service -->
<script src="assets/ng/Service/serviceFile.js"></script>

<!-- Data Table -->
<script src="https://cdn.jsdelivr.net/npm/angular-bootstrap-toggle@0.4.0/dist/angular-bootstrap-toggle.min.js"></script>

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script
src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->

<!-- Google Font -->
<link rel="stylesheet"

href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,
600,700,300italic,400italic,600italic">
</head>

<body class="hold-transition" ng-controller="Login">
<div class="login-page">

</div>
<div class="bg-text">
<div class="bg-text-container">
<div class="login-logo">

</div>
<!-- /.login-logo -->
<div class="login-box-body">
<p class="login-box-msg">Sign in here</p>
<form ng-submit="ProsesLogin()">
<div class="form-group has-feedback">
<input type="text" class="form-control" placeholder="Username"
required ng-model="DataInput.Username">

```

```

        <span class="glyphicon glyphicon-envelope form-control-
feedback"></span>
    </div>
    <div class="form-group has-feedback">
        <input type="password" class="form-control"
placeholder="Password" ng-model="DataInput.Password"
minlength="4" required>

        <span class="glyphicon glyphicon-lock form-control-
feedback"></span>
    </div>
    <div class="form-group has-feedback" style="text-align: right;">
        <button type="submit" class="btn btn-primary btn-flat">Sign
In</button>
    </div>
</form>
</div>
</div>
</div>
</div>
<!-- jQuery 3 -->

<!-- Bootstrap 3.3.7 -->
<script
src="assets/bower_components/bootstrap/dist/js/bootstrap.min.js"></script>
>
<script
src="assets/bower_components/ngSweetAlert/SweetAlert.min.js"></script>
>
<script
src="assets/bower_components/sweetalert/dist/sweetalert.min.js"></script>
>
<script src="plugins/iCheck/ichack.min.js"></script>
<script>
$(function () {
    $('input').iCheck({
        checkboxClass: 'icheckbox_square-blue',
        radioClass: 'iradio_square-blue',
        increaseArea: '20%' /* optional */
    });
});
</script>
</body>

</html>

```

## 2. Profile

### a. Controller

```
(function (angular) {
  'use strict'
  angular.module('BaseProfile',
    ['BaseProfileDirectives',
    'JadwalDirectives', 'KemajuanStudiDirective', 'MahasiswaWaliDirective'])
    .controller("BaseProfileController", function ($scope,
BaseProfileService, Jadwal, KhsmService, $window, SweetAlert,
WaliMahasiswa) {
      $scope.DatasHome = {};
      $scope.DatasJadwal = [];
      $scope.Khsm = {};
      $scope.Input = {};
      $scope.TahunAkademik = {};
      $scope.Testing;
      $scope.RoleMahasiswa = false;
      $scope.RolePegawai = false;
      $scope.DatasJadwal = [];
      $scope.ListMonitoring = [];
      $scope.DataLooping = [{ semester: '1' }, { semester: '2' }, { semester:
'3' }, { semester: '4' }, { semester: '5' }, { semester: '6' }, { semester: '7' }, {
semester: '8' }];
      $scope.Input.npm =
$window.sessionStorage.getItem("Username");
      BaseProfileService.get().then(response => {
        if (response.Role == "Mahasiswa") {
          BaseProfileService.getTA().then(response => {
            $scope.TahunAkademik = response;
          })
          $scope.RoleMahasiswa = true;
          $scope.DatasHome = angular.copy(response);
          Jadwal.getJadwalKuliah($scope.DatasHome).then(response
=> {
            $scope.DatasJadwal = response;
          }, error => {
          });
          KhsmService.get($scope.Input).then(response => {
            $scope.Khsm.Datas = [];
            var b = response.data;
            angular.forEach(b.Datas, function (value, key) {
              $scope.Khsm.Datas.push(angular.copy(value));
              $scope.Khsm.IPK = b.IPK;
              $scope.Khsm.SKS = b.SKS;
            });
            if ($scope.Khsm.IPK == 0)
```

```

        SweetAlert.swal("Kemajuan Studi Anda Masih Kosong,
        Segera melapor ke BAAK!!!!");
    }, error => {
        var a = error;
    });
    $scope.peringatan="";
    WaliMahasiswa.getList().then(response =>{
        $scope.ListMonitoring = response;

        if(parseInt($scope.ListMonitoring.data[0].SMT_Tempuh)<=4           ||
        (parseInt($scope.ListMonitoring.data[0].SKS_Lulus)<30           &&
        parseInt($scope.ListMonitoring.data[0].IPK<1)))){
            $scope.peringatan = $scope.ListMonitoring.pesan + " 1";
        }else
        if(parseInt($scope.ListMonitoring.data[0].SMT_Tempuh)<=6           &&
        parseInt($scope.ListMonitoring.data[0].SKS_Lulus)<50){
            $scope.peringatan = $scope.ListMonitoring.pesan + " 2";
        }else
        if(parseInt($scope.ListMonitoring.data[0].SMT_Tempuh)<=8           ||
        (parseInt($scope.ListMonitoring.data[0].SKS_Lulus)<700           &&
        parseInt($scope.ListMonitoring.data[0].IPK<2)))){
            $scope.peringatan = $scope.ListMonitoring.pesan + " 3";
        }else
        if(parseInt($scope.ListMonitoring.data[0].SMT_Tempuh)<=10           &&
        parseInt($scope.ListMonitoring.data[0].SKS_Lulus)<90){
            $scope.peringatan = $scope.ListMonitoring.pesan + "
        TERAKHIR";
        }else
        if(parseInt($scope.ListMonitoring.data[0].SMT_Tempuh)<=12           &&
        parseInt($scope.ListMonitoring.data[0].SKS_Lulus)<100){
            $scope.peringatan = $scope.ListMonitoring.pesan + "
        TERAKHIR";
        }

    });

    }else{
        BaseProfileService.getTA().then(response => {
            $scope.TahunAkademik = response;
            Jadwal.getJadwalDosen().then(response => {
                $scope.DatasJadwal = response;
            })
        })
        $scope.DatasHome = angular.copy(response);
        $scope.RolePegawai = true;
    }
}

```

```
});
```

```
});
```

```
))(window.angular);
```

**b. View**

```
<div id="loading-bar-container"></div>
```

```
<div ng-if="RoleMahasiswa==true">
```

```
  <section class="row content horizontal-scroll">
```

```
    <div class="col-md-3" style="padding: 2px;">
```

```
      <div class="box-body">
```

```
        <div class="box-blur">
```

```
          <div class="box-header with-border">
```

```
            <h3 class="box-title">Base Profile</h3>
```

```
          </div>
```

```
            <h3 class="profile-username text-center">{{DatasHome.nmmhs}}</h3>
```

```
            <h3 class="profile-username text-center">{{peringatan}}</h3>
```

```
            <p class="text-muted text-center"><b>{{DatasHome.npm}}</b></p>
```

```
            <p class="text-muted text-center">{{DatasHome.nmps}}&nbsp;{{DatasHome.jenjang1}}</p>
```

```
            <hr>
```

```
            <strong><i class="fa fa-map-marker margin-r-5"></i>Tahun Akademik</strong>
```

```
            <p class="text-muted">{{TahunAkademik.thakademik}}&nbsp;{{TahunAkademik.gg}}</p>  
            <hr />
```

```
            <strong><i class="fa fa-map-marker margin-r-5"></i>IPK</strong>
```

```
            <p class="text-muted">{{Khsm.IPK}}</p>  
            <hr />
```

```
            <strong><i class="fa fa-map-marker margin-r-5"></i>SKS Tempuh</strong>
```

```
            <p class="text-muted">{{Khsm.SKS}}</p>  
            <hr />
```

```
            <strong><i class="fa fa-map-marker margin-r-5"></i>Dosen Wali</strong>
```

```
            <p class="text-muted">{{DatasHome.nmdsn}}</p>  
            <hr />
```



```
                    <strong><i      class="fa      fa-map-marker      margin-r-5"></i>Kaprodi</strong>
                    <p class="text-muted">{{DatasHome>Nama}}</p>
```

```
</div>
```

```
</div>
```

```
<div class="box-body">
  <div class="box-blur">
    <div class="box-header">
      <h3 class="box-title">Info Detail</h3>
    </div>
    <hr>
    <strong><i class="fa fa-map-marker margin-r-5"></i>Tempat
Tanggal Lahir</strong>
    <p
                                class="text-muted">{{DatasHome.tmlhr}},
{{DatasHome.tglhr}}</p>
    <hr>
    <strong><i class="fa fa-book margin-r-5"></i>Agama</strong>
    <p class="text-muted">
      {{DatasHome.agama}}
    </p>
    <hr>
    <strong><i class="fa fa-book margin-r-5"></i>Pendidikan
Terakhir</strong>
    <p class="text-muted">
      {{DatasHome.pendidikan}}
    </p>
    <hr>
    <strong><i      class="fa      fa-map-marker      margin-r-5"></i>Alamat</strong>
    <p class="text-muted">{{DatasHome.almt}}</p>
    <hr>
    <strong><i      class="fa      fa-map-marker      margin-r-5"></i>Telp</strong>
    <p class="text-muted">{{DatasHome.notlp}}</p>
    <hr>
    <strong><i class="fa fa-pencil margin-r-5"></i> Skills</strong>
  </div>
</div>
<div class="box-body">
  <div class="box-blur">
    <div class="box-header with-border">
      <h3 class="box-title">Info Sekolah Asal</h3>
    </div>
```

```

        <strong><i class="fa fa-map-marker margin-r-5"></i>Nama
Sekolah</strong>
        <p class="text-muted">{{DatasHome.nmsmu}}</p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Jurusan</strong>
        <p class="text-muted">
            {{DatasHome.jursmu}}
        </p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Alamat</strong>
        <p class="text-muted">
            {{DatasHome.kabsmu}}, Provinsi {{DatasHome.provsmu}}
        </p>
        <hr>
        <strong><i class="fa fa-map-marker margin-r-5"></i>Asal
Sekolah</strong>
        <p class="text-muted">{{DatasHome.nmsmu}}</p>
    </div>
</div>
<div class="col-md-9" style="padding: 0;">
    <div class="nav-tabs-custom" style="margin: 5px;">
        <ul class="nav nav-tabs">
            <li class="active"><a href="" data-toggle="tab" data-
target="#activity">Kemajuan Studi</a></li>
            <li><a href="" data-toggle="tab" data-
target="#timeline">Jadwal Kuliah </a></li>
        </ul>
        <div class="tab-content">
            <div class="active tab-pane" id="activity">
                <ul class="timeline timeline-inverse" ng-repeat="loop in
DataLooping">
                    <li class="time-label" style="color:white;">
                        <div class="time-label-container">
                            <span>
                                Semester {{loop.semester}}
                            </span>
                        </div>
                    </li>
                    <li ng-repeat="item in Khsm.Datas | filter: { smt:
loop.semester }">
                        <i class="fa fa-item bg-blue"></i>
                        <div class="timeline-item" ng-if="item.ket=='L'">
                            <span class="time">SKS: &nbsp;&nbsp;{{item.sks}}
&nbsp;&nbsp;|&nbsp;&nbsp;

```

Keterangan:

```
<div class="timeline">  
    <div class="timeline-header"></div>  
    <div class="timeline-item" ng-if="item.ket!='L'"  
        style="background-color: rgb(214, 18, 18); color:white;">  
        <span class="time" style="color:white;">SKS:  
            <br/>{{item.sks}} </span>  
    </div>  
</div>
```

```

        </div>
    </li>
    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Selasa'
}">

        <i class="fa fa-item bg-blue"></i>

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>
                {{item.wm}} s/d
                {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}}</h3>
        </div>
    </li>
</ul>
<ul class="timeline timeline-inverse">
    <li class="time-label">
        <div class="time-label-container">
            <span >
                Rabu
            </span>
        </div>
    </li>
    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Rabu'
}">

        <i class="fa fa-item bg-blue"></i>

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>
                {{item.wm}} s/d
                {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}}</h3>
        </div>
    </li>
</ul>
<ul class="timeline timeline-inverse">
    <li class="time-label">
        <div class="time-label-container">
            <span >
                Kamis
            </span>
        </div>
    </li>
    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Kamis'
}">

        <i class="fa fa-item bg-blue"></i>

```

```

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
                {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}}</h3>
        </div>
    </li>
</ul>
<ul class="timeline timeline-inverse">
    <li class="time-label">
        <div class="time-label-container">
            <span >
                Jumat
            </span>
        </div>
    </li>
    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Jumat'
}">
        <i class="fa fa-item bg-blue"></i>

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
                {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}}</h3>
        </div>
    </li>
</ul>
<ul class="timeline timeline-inverse">
    <li class="time-label">
        <div class="time-label-container">
            <span >
                Sabtu
            </span>
        </div>
    </li>
    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Sabtu'
}">
        <i class="fa fa-item bg-blue"></i>

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
                {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}}</h3>
        </div>

```

```

        </li>
    </ul>
</div>

<div class="tab-pane" id="settings">

    </div>
</div>
</div>
</div>
</section>
</div>

<!-- View Pegawai -->
<div ng-if="RolePegawai==true">
    <section class="row content horizontal-scroll">
        <div class="col-md-3" style="padding: 2px;">
            <div class="box-body">
                <div class="box-blur">
                    <div class="box-header">
                        <h3 class="box-title">Base Profile</h3>
                    </div>
                    <hr>
                    <h3 class="profile-username text-center">{{DatasHome>Nama}}</h3>
                    <h4 class="text-center">{{DatasHome>NIK}}</h4>
                    <hr>
                    <strong><i class="fa fa-map-marker margin-r-5"></i>Tahun
Akademin</strong>
                    <p class="text-muted">{{TahunAkademik>.thakademik}}&nbsp;{{TahunAkademik>.gg}}</p>
                    <hr>
                    <strong><i class="fa fa-map-marker margin-r-5"></i>Status</strong>
                    <p class="text-muted">{{DatasHome>.Status}}</p>
                    <hr>

                </div>
            </div>
            <div class="box-body">

                <div class="box-blur">
                    <div class="box-header">
                        <h3 class="box-title">Info Pegawai</h3>
                    </div>
                    <hr>

```

```

        <strong><i class="fa fa-book margin-r-5"></i>NIDN</strong>
        <p class="text-muted">
            {{DatasHome.nidn}}
        </p>
        <hr>
        <strong><i class="fa fa-map-marker margin-r-5"></i>Tempat
Tanggal Lahir</strong>
        <p class="text-muted">{{DatasHome.TempatLahir}},
{{DatasHome.TanggalLahir}}</p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Alamat</strong>
        <p class="text-muted">
            {{DatasHome.Alamat}}
        </p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Kontak</strong>
        <p class="text-muted">
            {{DatasHome.Kontak}}
        </p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Tahun
Masuk</strong>
        <p class="text-muted">
            {{DatasHome.TahunMasuk}}
        </p>
        <strong><i class="fa fa-book margin-r-5"></i>Pendidikan
Terakhir</strong>
        <p class="text-muted">
            {{DatasHome.pendakhir}}
        </p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Spesifikasi
Dosen</strong>
        <p class="text-muted">
            {{DatasHome.spekdsn}}
        </p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>Status
Dosen</strong>
        <p class="text-muted">
            {{DatasHome.stdsn}}
        </p>
        <hr>
        <strong><i class="fa fa-book margin-r-5"></i>SK
Dosen</strong>
        <p class="text-muted">

```

```

        {{DatasHome.noskstimik}}
    </p>
    <hr>
    <strong><i class="fa fa-book margin-r-5"></i>Tanggal SK
Dosen</strong>
    <p class="text-muted">
        {{DatasHome.tgskstimik}}
    </p>
    <hr>
</div>
</div>
<div class="col-md-9" style="padding: 2px;">
    <div class="nav-tabs-custom">
        <ul class="nav nav-tabs">
            <li class="active"><a href="" data-toggle="tab" data-
target="#activity">Jadwal Mengajar</a></li>
            <!-- <li><a href="" data-toggle="tab" data-target="#nilai">Nilai
Mahasiswa</a></li> -->
            <!-- <li><a href="" data-toggle="tab" data-
target="#wali">Mahasiswa Wali</a></li> -->
            <!-- <li><a href="" data-toggle="tab" data-
target="#settings">Settings</a></li> -->
        </ul>
        <div class="tab-content">
            <div class="active tab-pane" id="activity">
                <ul class="timeline timeline-inverse">
                    <li class="time-label">
                        <span class="bg-red">
                            Senin
                        </span>
                    </li>
                    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Senin'
}">
                        <i class="fa fa-item bg-blue"></i>

                        <div class="timeline-item">
                            <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
                                {{item.ws}} | Ruang: {{item.ruangan}}</span>
                            <h3 class="timeline-header">{{item.nmmk}} | Kelas :
{{item.kelas}}</h3>
                        </div>
                    </li>
                </ul>
            </div>
        </div>
    </div>

```



```

        <li class="time-label">
            <span class="bg-red">
                Selasa
            </span>
        </li>
        <li ng-repeat="item in DatasJadwal | filter: { hari: 'Selasa'
    }">

        <i class="fa fa-item bg-blue"></i>

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>

{{item.wm}} s/d

            {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}} | Kelas :
{{item.kelas}}</h3>
        </div>
        </li>
    </ul>
    <ul class="timeline timeline-inverse">
        <li class="time-label">
            <span class="bg-red">
                Rabu
            </span>
        </li>
        <li ng-repeat="item in DatasJadwal | filter: { hari: 'Rabu'
    }">

        <i class="fa fa-item bg-blue"></i>

        <div class="timeline-item">
            <span class="time"><i class="fa fa-clock-o"></i>

{{item.wm}} s/d

            {{item.ws}} | Ruang: {{item.ruangan}}</span>
            <h3 class="timeline-header">{{item.nmmk}} | Kelas :
{{item.kelas}}</h3>
        </div>
        </li>
    </ul>
    <ul class="timeline timeline-inverse">
        <li class="time-label">
            <span class="bg-red">
                Kamis
            </span>
        </li>
        <li ng-repeat="item in DatasJadwal | filter: { hari: 'Kamis'
    }">

        <i class="fa fa-item bg-blue"></i>

```

```

<div class="timeline-item">
  <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
    {{item.ws}} | Ruang: {{item.ruangan}}</span>
    <h3 class="timeline-header">{{item.nmmk}} | Kelas :
{{item.kelas}}</h3>
  </div>
</li>
</ul>
<ul class="timeline timeline-inverse">
  <li class="time-label">
    <span class="bg-red">
      Jumat
    </span>
  </li>
  <li ng-repeat="item in DatasJadwal | filter: { hari: 'Jumat'
}">
    <i class="fa fa-item bg-blue"></i>

    <div class="timeline-item">
      <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
        {{item.ws}} | Ruang: {{item.ruangan}}</span>
        <h3 class="timeline-header">{{item.nmmk}} | Kelas :
{{item.kelas}}</h3>
      </div>
    </li>
  </ul>
  <ul class="timeline timeline-inverse">
    <li class="time-label">
      <span class="bg-red">
        Sabtu
      </span>
    </li>
    <li ng-repeat="item in DatasJadwal | filter: { hari: 'Sabtu'
}">
      <i class="fa fa-item bg-blue"></i>

      <div class="timeline-item">
        <span class="time"><i class="fa fa-clock-o"></i>
{{item.wm}} s/d
          {{item.ws}} | Ruang: {{item.ruangan}}</span>
          <h3 class="timeline-header">{{item.nmmk}} | Kelas :
{{item.kelas}}</h3>
        </div>
      </li>
    </ul>

```

```

        </li>
    </ul>
</div>
<div class="tab-pane" id="nilai">
    <h3>Ongoing</h3>
</div>

<div class="tab-pane" id="wali">
</div>
</div>
</div>
</section>
</div>

```

### 3. Pengajuan KRS

#### a. Controller

```

(function (angular) {
    'use strict'
    angular.module('PengajuanKRS',                ['PengajuanKRSDirective',
        'JadwalDirectives', 'MahasiswaDirectives'])

    .filter('total', function () {
        return function (input, property) {
            if (input.length !== undefined) {
                var i = input.length;
                var total = 0;
                while (i--)
                    total += parseInt(input[i][property]);
                return total;
            }
        }
    })

    .controller('PengajuanKRSController',        function ($scope,
        PengajuanService, $window, Jadwal, $filter, $state, SweetAlert,
        MahasiswaService, $sce) {

        // setTimeout(x => {
        $scope.GetValue;
        $scope.jmsms;
        $scope.showJadwal = false;
        $scope.showJadwal1 = false;
        $scope.TemKrsmJadwal = false;
        $scope.KrsmJadwal = false;
        $scope.showKrsm = false;

```

```

$scope.DatasJadwal = [];
$scope.DatasKrsm = [];
$scope.CariJadwal = "";
$scope.Kelas = "";
$scope.Tombol = true;
$scope.BatasReg = false;
$scope.DatasKrsm.DetailKrsm = [];
$scope.jmsks;

```

```

PengajuanService.get().then(response => {
    $scope.jmsks = 0;

    if (response.set == 'Jadwal') {
        $scope.showJadwal = true;
        $scope.showKrsm = false;
        $scope.TemKrsmJadwal = false;
        $scope.KrsmJadwal = false;
        $scope.showJadwal1 = true;
        $scope.BatasReg = false;
        $scope.DatasJadwal = response.data.data;
        $scope.GetValue = angular.copy($scope.DatasJadwal[0]);
        if (response.mahasiswa.kurikulum == "2018") {
            if (parseInt(response.semester) + 1 == 2) {
                var agama;
                if (response.mahasiswa.agama == "Islam") {
                    agama = "Islam";
                } else {
                    agama = "Kristen";
                }
                angular.forEach($scope.DatasJadwal, function (value) {
                    if (parseInt(value.smt) == 2) {
                        if (value.nmmk == "PENDIDIKAN AGAMA") {
                            if (agama == "Islam" && value.kelas == "A") {
                                $scope.jmsks +=
                                parseInt(angular.copy(value.sks));
                                value.status = false;
                                value.ngBinding = "success";
                                $scope.DataTampung.push(value);
                            } else if (agama == "Kristen" && value.kelas ==
                                "B") {
                                    $scope.jmsks +=
                                    parseInt(angular.copy(value.sks));
                                    value.status = false;
                                    value.ngBinding = "success";
                                    $scope.DataTampung.push(value);
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

    }
    } else if(response.mahasiswa.kelas == value.kelas) {
        $scope.jmsks +=
parseInt(angular.copy(value.sks));
        value.status = false;
        value.ngBinding = "success";
        $scope.DataTampung.push(value);
    }
    // $scope.Tombol = false;
}
})
} else {
    angular.forEach($scope.DatasJadwal, function (value) {
        if (parseInt(value.smt) == response.semester+1 &&
response.mahasiswa.kelas == value.kelas) {
            $scope.jmsks += parseInt(angular.copy(value.sks));
            value.status = false;
            value.ngBinding = "success";
            $scope.DataTampung.push(value);
        }
    })
}
}
} else if (response.set == 'TemKrsm') {

    $scope.showJadwal = false;
    $scope.showKrsm = false;
    $scope.TemKrsmJadwal = true;
    $scope.KrsmJadwal = false;
    $scope.showJadwal1 = false;
    $scope.BatasReg = false;
    $scope.DatasTemKrsm = response.data.data;
    SweetAlert.swal(
    {
        title: 'Information!',
        type: 'info',
        text: 'Pengajuan KRS anda sudah di <strong>' +
$scope.DatasTemKrsm.TemKrsm[0].status + '</strong>',
        html: true
    });
    Jadwal.get().then(response => {
        var a = JSON.parse(response.data);

        $scope.GetValue = angular.copy(a[0]);

```

```

MahasiswaService.get($scope.DatasTemKrsm.TemKrsm[0].npm).then(res
ponse => {
    a = $filter('filter')(a, function (value) {
        return (value.kurikulum ===
response.data[0].kurikulum || value.kurikulum === 'ALL') && value.kdps ===
response.data[0].kdps;
    });
    angular.forEach($scope.DatasTemKrsm.TemDetailKrsm,
function (value, key) {
        $scope.jmsks += parseInt(value.sks);
        angular.forEach(a, function (value1, key1) {
            if (value.kmk == value1.kmk && value.kelas ==
value1.kelas) {
                value1.status = true;
                value1.ngBinding = "success";
                value.hari = value1.hari;
                value.ws = value1.ws;
                value.wm = value1.wm;
            }
        });
    });
    $scope.DatasJadwal = a;
})

});

} else if (response.set == 'Krsm') {
    $scope.DatasKrsm = response.data.data;
    angular.forEach($scope.DatasKrsm.DetailKrsm, function
(value, key) {
        $scope.jmsks += parseInt(value.sks);
    });
    $scope.GetValue = angular.copy($scope.DatasJadwal[0]);
    $scope.showJadwal = false;
    $scope.showKrsm = true;
    $scope.TemKrsmJadwal = false;
    $scope.KrsmJadwal = false;
    $scope.showJadwal1 = false;
    $scope.BatasReg = false;
    // SweetAlert.swal("KRS Anda telah selesai di proses");
} else if (response.set == 'Daftar Ulang') {
    $scope.showKrsm = false;
    $scope.showJadwal = false;
    $scope.TemKrsmJadwal = false;
    $scope.showJadwal1 = false;

```

```

$scope.KrsmJadwal = false;
$scope.BatasReg = false;
SweetAlert.swal("Anda Belum Terdaftar Silahkan hubungi
bagian Program Studi");
} else if (response.set == 'BatasReg') {
    $scope.showKrsm = false;
    $scope.showJadwal = false;
    $scope.showJadwal1 = false;
    $scope.TemKrsmJadwal = false;
    $scope.KrsmJadwal = false;
    $scope.BatasReg = true;
    SweetAlert.swal("Registrasi KRS Telah Tutup, Silahkan
Hubungi Program Studi");
} else if (response.set == 'MulaiReg') {
    $scope.showKrsm = false;
    $scope.showJadwal = false;
    $scope.showJadwal1 = false;
    $scope.TemKrsmJadwal = false;
    $scope.KrsmJadwal = false;
    $scope.BatasReg = true;
    SweetAlert.swal("Registrasi KRS Belum di Buka");
} else {
    $scope.showKrsm = false;
    $scope.showJadwal = false;
    $scope.showJadwal1 = false;
    $scope.TemKrsmJadwal = false;
    $scope.KrsmJadwal = false;
    SweetAlert.swal("Data Jadwal Belum Tersedia Segera
Hubungi Bagian BAAK Untuk ketersediaan");
}
}, error => {
    console.log(error);
    if (error.data.message == "Session Habis") {
        SweetAlert.swal("Session Anda Habis Silahkan Login
kembali");
        window.location.href = "index.html";
    } else if (error.data.message == "Data Jadwal Belum Tersedia
Segera Hubungi Bagian BAAK Untuk ketersediaan") {
        SweetAlert.swal("Data Jadwal Belum Tersedia Segera
Hubungi Bagian BAAK Untuk ketersediaan");
    }
});

$scope.PilihItem = function (item) {
    if (item.ngBinding == "success") {
        alert("\nMatakuliah tersebut sudah dipilih");
    }
}

```

```

    } else {
      if ($scope.jmsks + parseInt(item.sks) <= 24) {
        SweetAlert.swal({
          title: "Confirmation!!!",
          text: "Anda Yakin ingin Menambah Matakuliah?",
          type: "info",
          showCancelButton: true,
          confirmButtonColor: "#0be7fb",
          confirmButtonText: "Yes, Ubah!",
          cancelButtonText: "No, Batal!",
          closeOnConfirm: false,
          closeOnCancel: true,
          showLoaderOnConfirm: true
        },
        function (isConfirm) {
          if (isConfirm) {
            $scope.DataKrs = {
              'thakademik': $scope.GetValue.thakademik,
              'gg': $scope.GetValue.gg,
              'npm':
$window.sessionStorage.getItem("Username"),
              'sms': "",
              'dsn_wali': "",
              'ketjur': "",
              'admakademik': "",
              'jmsks': $scope.jmsks,
              'tgkrsm': "",
              'kdps': $scope.GetValue.kdps
            };
            var data = {};
            data.krsm = $scope.DataKrs;
            data.DetailKrsm = item;
            PengajuanService.post(data).then(response => {
              SweetAlert.swal("Information!!!",
response.message, "success");
              $scope.jmsks +=
parseInt(angular.copy(item.sks));
              item.status = false;
              item.ngBinding = "success";

              $scope.DatasTemKrsm.TemDetailKrsm.push(item);
            }, error => {
              //console.log(error);
              $state.reload();
            });
          }
        });
      }
    }
  }

```



```

        } else {
            SweetAlert.swal("Cancelled", "Perubahan Password
di batalkan:", "error");
        }
    }
);

} else
    alert("\nTotal SKS yang ada pilih melebihi Batas");

}
}

$scope.DataTampung = [];
$scope.Pilih = function (item) {
    if (item.ngBinding == "success") {
        alert("\nMatakuliah tersebut sudah dipilih");
    } else {
        if ($scope.jmsks + parseInt(item.sks) <= 24) {
            $scope.jmsks += parseInt(angular.copy(item.sks));
            item.status = false;
            item.ngBinding = "success";
            $scope.DataTampung.push(item);
            $scope.Tombol = false;
        } else
            SweetAlert.swal("Info", "Total SKS yang ada pilih melebihi
Batas)", "error");
        }
    }

    $scope.HapusPilihan = function (item) {

        $scope.jmsks -= parseInt(angular.copy(item.sks));
        var index = $scope.DataTampung.findIndex(DataTampung =>
DataTampung.idjadwal == item.idjadwal);
        $scope.DataTampung.splice(index, 1);
        angular.forEach($scope.DatasJadwal, function (value, key) {
            if (value.idjadwal == item.idjadwal) {
                value.ngBinding = "";
            }
        });
    }

    $scope.Hapus = function (item) {
        SweetAlert.swal({
            title: "Confirmation!!!",

```

```

        text: "Anda Yakin ingin Menghapus Matakuliah?",
        type: "info",
        showCancelButton: true,
        confirmButtonColor: "red",
        confirmButtonText: "Hapus!",
        cancelButtonText: "Batal!",
        closeOnConfirm: false,
        closeOnCancel: true,
        showLoaderOnConfirm: true
    },
    function (isConfirm) {
        if (isConfirm) {
            $scope.DataKrs = {
                'thakademik': $scope.GetValue.thakademik,
                'gg': $scope.GetValue.gg,
                'npm': $window.sessionStorage.getItem("Username"),
                'sms': "",
                'dsn_wali': "",
                'ketjur': "",
                'admakademik': "",
                'jmsks': $scope.jmsks,
                'tgkrsm': "",
                'kdps': $scope.GetValue.kdps
            };
            var data = {};
            data.krsm = $scope.DataKrs;
            data.DetailKrsm = item;
            PengajuanService.delete(item).then(response => {
                SweetAlert.swal("Information!!!", "Item berhasil di
hapus", "success");
                var kmk = angular.copy(item.kmk);
                $scope.jmsks -= parseInt(angular.copy(item.sks));
                $scope.Data
                $scope.DatasTemKrsm.TemDetailKrsm
                var index = $scope.Data.findIndex(Data => Data.kmk
                == item.kmk);
                $scope.DatasTemKrsm.TemDetailKrsm.splice(index,
                1);
                angular.forEach($scope.DatasJadwal, function (value,
                key) {
                    if (value.kmk == item.kmk) {
                        value.ngBinding = "";
                    }
                });
            }, error => {
                console.log(error);
            });
        }
    }

```

```

        alert(response.message);
    });
    } else {
        SweetAlert.swal("Cancelled", "Perubahan Password di
batalan:)", "error");
    }
}
);

```

```

}
$scope.AjukanKRS = function () {
    if ($scope.DataTampung.length > 0) {
        SweetAlert.swal({
            title: "Confirmation!!!",
            text: "Anda akan melakukan pengajuan KRSM",
            type: "warning",
            showCancelButton: true,
            confirmButtonColor: "#0be7fb",
            confirmButtonText: "Yes, Ubah!",
            cancelButtonText: "No, Batal!",
            closeOnConfirm: false,
            closeOnCancel: false,
            showLoaderOnConfirm: true
        },
        function (isConfirm) {
            if (isConfirm) {
                $scope.Tombol = true;
                $scope.DataKrs = {
                    'thakademik': $scope.GetValue.thakademik,
                    'gg': $scope.GetValue.gg,
                    'npm':
$window.sessionStorage.getItem("Username"),
                    'sms': "",
                    'dsn_wali': "",
                    'ketjur': "",
                    'admakademik': "",
                    'jmsks': $scope.jmsks,
                    'tgkrsm': "",
                    'kdps': $scope.GetValue.kdps
                };
                var data = {};
                data.krsm = $scope.DataKrs;
                data.DetailKrs = $scope.DataTampung;

                PengajuanService.post(data).then(response => {

```



```

        </tr>
    </thead>
    <tbody style="height: 200px;">
        <tr ng-repeat="item in DatasJadwal | filter: { kelas: Kelas }
| filter: CariJadwal | orderBy : 'kelas'"
        ng-click="Pilih(item)">
            <td                                width="5%"
class="{{item.ngBinding}}">{{item.smt}}</td>
            <td                                width="10%"
class="{{item.ngBinding}}">{{item.kelas}}</td>
            <td class="{{item.ngBinding}}">{{item.nmmk}}</td>
            <td                                width="5%"
class="{{item.ngBinding}}">{{item.sks}}</td>
            <td class="{{item.ngBinding}}">{{item.ruangan}}</td>
            <td
class="{{item.ngBinding}}">{{item.hari}},&nbsp;{{item.wm}}-{{item.ws}}</td>
            <td class="{{item.ngBinding}}">{{item.dsn_saji}}</td>
        </tr>
    </tbody>
</table>
</div>

```

```

</div>
<div class="box-footer clearfix">
    <div class="box-header with-border">
        <button type="submit" class="btn btn-default"
disabled><b>Total SKS: {{jmsks}}</b></button>
        <div class="box-tools">
            <div class="input-group input-group-sm" style="width:
200px;">
                <input type="text" name="table_search" class="form-
control pull-right" placeholder="Show By Kelas"
                ng-model="Kelas">

                <div class="input-group-btn">
                    <button type="submit" class="btn btn-default"><i
class="fa fa-search"></i></button>
                </div>
            </div>
        </div>
    </div>
</div>
</div>
</div>

```

```

<div class="box" ng-show="showJadwal1">

```

```

<div class="box-header with-border">
  <h3 class="box-title">Daftar Matakuliah yang Dipilih</h3>
</div>
<div class="box-body">
  <div class="box-body table-responsive no-padding">
    <table datatable="ng" class="table table-bordered table-
hover">
      <thead>
        <tr>
          <th>Semester</th>
          <th>Kelas</th>
          <th>Matakuliah</th>
          <th>SKS</th>
          <th>Ruangan</th>
          <th>Waktu Kuliah</th>
          <th>Dosen Pengampuh</th>
          <th>Action</th>
        </tr>
      </thead>
      <tbody>
        <tr ng-repeat="item in DataTampung | groupBy:'smt'">
          <td>{{item.smt}}</td>
          <td>{{item.kelas}}</td>
          <td>{{item.nmmk}}</td>
          <td>{{item.sks}}</td>
          <td>{{item.ruangan}}</td>
          <td>{{item.hari}},&nbsp;{{item.wm}}-{{item.ws}}</td>
          <td>{{item.dsn_saji}}</td>
          <td><button type="button" class="btn btn-default" ng-
click="HapusPilihan(item)"><i
class="fa fa-remove"></i></button></td>
        </tr>
      </tbody>
    </table>
  </div>
</div>
<div class="box-footer clearfix">
  <button type="submit" class="btn btn-default" disabled><b>Total
SKS: {{jmsks}}</b></button>
  <button type="button" class="pull-right btn btn-default" id="Send"
ng-click="AjukanKRS()" ng-disabled= "Tombol">Ajukan KRS
    <i class="fa fa-arrow-circle-right"></i></button>
  </div>
</div>
</div>

```

```

<div ng-show="TemKrsmJadwal">
  <div class="box">
    <div class="box-header with-border">
      <h3 class="box-title">Pilih Matakuliah yang akan di kontrak
dengan Bijak</h3>
    </div>
    <div class="box-body">
      <div class="box-body table-responsive no-padding">
        <table datatable="ng" class="table table-bordered table-
hover">
          <thead>
            <tr>
              <th style="width: 10%">Semester</th>
              <th style="width: 10%">Kelas</th>
              <th>Matakuliah</th>
              <th style="width: 10%">SKS</th>
              <th style="width: 10%">Ruangan</th>
              <th>Waktu Kuliah</th>
              <th>Dosen Pengampuh</th>
            </tr>
          </thead>
          <tbody style="height: 200px;">
            <tr ng-repeat="item in DatasJadwal | filter: { kelas: Kelas }
| filter: CariJadwal | orderBy : 'kelas'" ng-click="PilihItem(item)">
              <td style="width: 10%"
class="{{item.ngBinding}}">{{item.smt}}</td>
              <td style="width: 10%"
class="{{item.ngBinding}}">{{item.kelas}}</td>
              <td class="{{item.ngBinding}}">{{item.nmmk}}</td>
              <td style="width: 10%"
class="{{item.ngBinding}}">{{item.sks}}</td>
              <td style="width: 10%"
class="{{item.ngBinding}}">{{item.ruangan}}</td>
              <td
class="{{item.ngBinding}}">{{item.hari}},&nbsp;{{item.wm}}-{{item.ws}}</td>
              <td class="{{item.ngBinding}}">{{item.dsn_saji}}</td>
            </tr>
          </tbody>
        </table>
      </div>
    </div>
    <div class="box-footer clearfix">
      <button type="submit" class="btn btn-default" disabled><b>Total
SKS: {{jmsks}}</b></button>
    </div>
  </div>

```

```

</div>
<div class="box">

    <div class="box-header with-border">
        <h3 class="box-title">Daftar Matakuliah Yang Telah
diajukan</h3>
    </div>
    <div class="box-body">
        <div class="box-body table-responsive no-padding">
            <table datatable="ng" class="table table-bordered table-
hover">
                <thead>
                    <tr>
                        <th width="10%">Semester</th>
                        <th width="10%">Kelas</th>
                        <th>Matakuliah</th>
                        <th width="10%">SKS</th>
                        <th>Dosen Ampu</th>
                        <th>Waktu Kuliah</th>
                        <th></th>
                    </tr>
                </thead>
                <tbody>
                    <tr ng-repeat="item in DatasTemKrsm.TemDetailKrsm">
                        <td width="10%">{{item.smt}}</td>
                        <td width="10%">{{item.kelas}}</td>
                        <td>{{item.nmmk}}</td>
                        <td width="10%">{{item.sks}}</td>
                        <td>{{item.dsnampu}}</td>
                        <td>{{item.hari}},&nbsp;{{item.wm}}-{{item.ws}}</td>
                        <td><button type="button" class="btn btn-default" ng-
click="Hapus(item)"><i
                                class="fa fa-remove"></i></button></td>
                    </tr>
                </tbody>
            </table>
        </div>

    </div>
    <div class="box-footer clearfix">
        <button type="submit" class="btn btn-default" disabled><b>Total
SKS: {{jmsks}}</b></button>
    </div>
</div>
</div>

```



```
<div ng-show="showKrsm">  
    <div class="box">  
        <div class="box-header with-border">  
            <h3 class="box-title">KRS Anda</h3>  
        </div>  
        <div class="box-body">  
            <div class="box-body table-responsive no-padding">  
                <table class="table table-bordered table-hover">  
                    <thead>  
                        <tr>  
                            <th width="3%">No</th>  
                            <th width="50%">Matakuliah</th>  
                            <th width="10%">Kelas</th>  
                            <th width="10%">Semester</th>  
                            <th>SKS</th>  
                            <th>Nilai</th>  
                            <th>NxSKS</th>  
                            <th>Keterangan</th>  
                        </tr>  
                    </thead>  
                    <tbody style="height: 500px;">  
                        <tr ng-repeat="item in DatasKrsm.DetailKrsm">  
                            <td width="3%">{{ $index + 1 }}</td>  
                            <td width="50%">{{ item.kmk }}&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&{{ item.nmmk }}</td>  
                            <td width="10%">{{ item.kelas }}</td>  
                            <td width="10%">{{ item.smt }}</td>  
                            <td>{{ item.skss }}</td>  
                            <td>{{ item.nhuruf }}</td>  
                            <td>{{ item.nxskss }}</td>  
                            <td>{{ item.ket }}</td>  
                        </tr>  
  
                    </tbody>  
                    <tfoot>  
                        <tr>  
                            <td colspan="4" width="73%"><b>Total</b></td>  
                            <td>{{ DatasKrsm.DetailKrsm|total:'skss' }}</td>  
                            <td></td>  
                            <td>  
                                if="DatasKrsm.DetailKrsm|total:'nxskss'!=NaN">{{ DatasKrsm.DetailKrsm|tot  
al:'nxskss'}}</td>  
                            <td></td>  
                            <td></td>  
                        </tr>
```

```

        </tfoot>

    </table>
</div>
</div>
<div class="box-footer clearfix">
    <button type="submit" style="margin: 5px;" class="btn btn-
default" disabled><b>IP Semester Anda:
{{{(Dataskrsm.DetailKrsms|total:'nxsks')}}
(Dataskrsm.DetailKrsms|total:'sks')) | number:2}}</b></button>
</div>
</div>
</div>
</section>

```

#### 4. Persetujuan KRS

##### a. Controller

```

(function (angular) {
    'use strict'
    angular.module('ApprovedKRS', ['ApprovedDirectives',
    'JadwalDirectives', 'KemajuanStudiDirective'])
        .controller('ApprovedKRSController', function ($scope,
    ApprovedService, $window, Jadwal, $filter, $http, KhsmService,
    SweetAlert, DTOptionsBuilder, DTColumnDefBuilder) {
        $scope.ShowKeuangan = false;
        $scope.ShowWali = false;
        var DataStatus = {};
        $scope.DatasJadwal = [];
        $scope.SearchText = "";
        $scope.CariKemajuan = "";
        $scope.CariJadwal = "";
        $scope.Kelas = "";
        $scope.TampilJadwal = false;
        $scope.Datatampung = [];
        $scope.jmsks = 0;
        $scope>NamaSelected = "";
        $scope.NPMSelected = "";
        $scope.KemajuanStudi = [];
        $scope.Tombol = false;
        $scope.Histori = [];
        DataStatus.status = $window.sessionStorage.getItem("SetStatus");
        $scope.dtOptions =
        DTOptionsBuilder.newOptions().withPaginationType('full_numbers').withDi
        splayLength(2);
        $scope.dtColumnDefs = [
            DTColumnDefBuilder.newColumnDef(0),

```

```

        DTColumnDefBuilder.newColumnDef(1).notVisible(),
        DTColumnDefBuilder.newColumnDef(2).notSortable()
    ];
    if (DataStatus.status == "Dosen Wali" || DataStatus.status ==
"Prodi") {
        $scope.ShowKeuangan = false;
        $scope.ShowWali = true;

        Jadwal.get().then(response => {
            $scope.DatasJadwal = JSON.parse(response.data);
        }, error => {
            console.log(error);
        });
        ApprovedService.get(DataStatus).then(response => {
            $scope.jmsks = 0;
            $scope.TemKrsm = response.data.TemKrsm;
            $scope.Histori = response.data.Histori;
        }, error => {
            alert(error.data);
        });
    } else {
        $scope.ShowKeuangan = true;
        $scope.ShowWali = false;
        ApprovedService.get(DataStatus).then(response => {
            $scope.TemKrsm = response.data.TemKrsm;
            $scope.Histori = response.data.Histori;
        }, error => {
            alert(error.data);
        });
    }
}

$scope.ShowJadwalMahasiswa = function (item) {
    $scope.jmsks = 0;
    $scope>NamaSelected = item.nmmhs;
    $scope.NPMSelected = item.npm;
    $scope.TampilJadwal = true;
    $scope.DatasTampil = angular.copy($scope.DatasJadwal);
    $scope.Datatampung = item.detailTemKrsm[0];
    var data = {};
    data.npm = item.npm;
    KhsmService.get(data).then(response => {
        var b = response.data;
        $scope.KemajuanStudi.Datas = [];
        angular.forEach(b.Datas, function (value, key) {
            $scope.KemajuanStudi.Datas.push(angular.copy(value));
        });
    });
}

```

```

        $scope.KemajuanStudi.IPK = b.IPK;
        $scope.KemajuanStudi.SKS = b.SKS;
    });
}, error => {
    console.log(error);
})

angular.forEach($scope.Datatampung, function (value, key) {
    $scope.jmsks += parseInt(value.sks);
    angular.forEach($scope.DatasTampil, function (value1, key1) {
        if (value.kmk == value1.kmk && value.kelas == value1.kelas)
    {
        value.ruangan = value1.ruangan;
        value.hari = value1.hari;
        value.wm = value1.wm;
        value.ws = value1.ws;
        value.dsn_saji = value1.dsn_saji;
    }
    });
});
}
$scope.Approved = function (item) {
    $scope.Tombol = true;
    if ($scope.jmsks != NaN) {
        SweetAlert.swal({
            title: "Anda Yakin?",
            text: "Anda akan Menyetujui KRSM Mahasiswa: " +
item.nmmhs,
            type: "warning",
            showCancelButton: true,
            confirmButtonColor: "#0be7fb",
            confirmButtonText: "Yes, Approved!",
            cancelButtonText: "No, Batal!",
            closeOnConfirm: false,
            closeOnCancel: false,
            showLoaderOnConfirm: true
        },
        function (isConfirm) {
            if (isConfirm) {
                ApprovedService.put(item).then(response => {
                    var index = $scope.TemKrsm.findIndex(TemKrsm
=> TemKrsm.Id == item.Id);
                    $scope.TemKrsm.splice(index, 1);
                    SweetAlert.swal("Approved!", "Your proses krsm has
been approved.", "success");
                    $scope.Histori.push(angular.copy(item));

```



```

        <li class="active"><a href="" data-target="#persetujuan"
data-toggle="tab">Persetujuan</a></li>
        <li><a href="" data-target="#histori" data-
toggle="tab">Histori</a></li>
    </ul>
    <div class="tab-content">
        <div class="tab-pane active" id="persetujuan">
            <div class="box-body table-responsive no-padding">
                <table datatable="ng" class="table table-bordered
table-hover">
                    <thead>
                        <tr>
                            <th>Nama</th>
                            <th>NPM</th>
                            <th>Semester</th>
                            <th>Dosen Wali</th>
                            <th>Tanggal</th>
                            <th>Action</th>
                        </tr>
                    </thead>
                    <tbody>
                        <tr ng-repeat="item in TemKrsm | filter :
CariJadwal">
                            <td>{{item.nmmhs}}</td>
                            <td>{{item.npm}}</td>
                            <td>{{item.sms}}</td>
                            <td>{{item.dsn_wali}}</td>
                            <td>{{item.tgkrsm}}</td>
                            <td><button type="button" class="pull-right
btn btn-default" id="Send"
                                ng-click="Approved(item)" ng-disabled
= 'Tombol'>Approved
                                <i class="fa fa-arrow-circle-
right"></i></button>
                            </td>
                        </tr>
                    </tbody>
                </table>
            </div>
        </div>
        <div class="tab-pane" id="histori">
            <div class="box-body table-responsive no-padding">
                <table datatable="ng" class="table table-bordered
table-hover">
                    <thead>5

```

```

        <tr>
            <th>Nama</th>
            <th>NPM</th>
            <th>Semester</th>
            <th>Dosen Wali</th>
            <th>Tanggal</th>
            <th>Status</th>
        </tr>
    </thead>
    <tbody>
        <tr ng-repeat="item in Histori">
            <td>{{item.nmmhs}}</td>
            <td>{{item.npm}}</td>
            <td>{{item.sms}}</td>
            <td>{{item.dsn_wali}}</td>
            <td>{{item.tgkrsm}}</td>
            <td>{{item.status}}</td>
        </tr>
    </tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

<div class="box" ng-show="ShowWali">
    <div id="loading-bar-container"></div>
    <div class="box-header with-border">
        <h3 class="box-title">Pengajuan KRS Mahasiswa</h3>
    </div>
    <div class="box-body">
        <div class="nav-tabs-custom">
            <ul class="nav nav-tabs">
                <li class="active"><a href="" data-target="#persetujuanWali"
data-toggle="tab">Persetujuan</a></li>
                <li><a href="" data-target="#historiWali" data-
toggle="tab">Histori</a></li>
            </ul>
        </div>
    </div>
</div>

```

```

</ul>
<div class="tab-content">
  <div class="tab-pane active" id="persetujuanWali">
    <div class="box-body table-responsive no-padding">
      <table datatable="ng" class="table table-bordered table-
hover">
        <thead>
          <tr>
            <th>Nama</th>
            <th>NPM</th>
            <th>Kelas</th>
            <th>Semester</th>
            <th>Dosen Wali</th>
            <th>Tanggal</th>
            <th width=15%>Action</th>
          </tr>
        </thead>
        <tbody>
          <tr ng-repeat="item in TemKrsm | filter : SearchText">
            <td>{{item.nmmhs}}</td>
            <td>{{item.npm}}</td>
            <td>{{item.kelas}}</td>
            <td>{{item.sms}}</td>
            <td>{{item.dsn_wali}}</td>
            <td>{{item.tgkrsm}}</td>
            <td>
              <button type="submit" class="btn btn-danger" ng-
click="ShowJadwalMahasiswa(item)">Tampil
              Detail</button>
              <button type="button" class="pull-right btn btn-
default" id="Send"
              ng-click="Approved(item)">Approved
              <i class="fa fa-arrow-circle-right"></i></button>
            </td>
          </tr>
        </tbody>
      </table>
    </div>
  </div>
  <div class="tab-pane" id="historiWali">
    <div class="box-body table-responsive no-padding">
      <table datatable="ng" class="table table-bordered table-
hover">
        <thead>
          <tr>
            <th>Nama</th>

```



```

        <th>NPM</th>
        <th>Semester</th>
        <th>Dosen Wali</th>
        <th>Tanggal</th>
        <th>Status</th>
    </tr>
</thead>
<tbody>
    <tr ng-repeat="item in Histori">
        <td>{{item.nmmhs}}</td>
        <td>{{item.npm}}</td>
        <td>{{item.sms}}</td>
        <td>{{item.dsn_wali}}</td>
        <td>{{item.tgkrsm}}</td>
        <td>{{item.status}}</td>
    </tr>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
<div ng-show="TampilJadwal">
    <div class="box">
        <div class="box-header with-border">
            <h3 class="box-title">Kemajuan Studi
            &nbsp;<b>{{NamaSelected}}&nbsp;<b>|&nbsp;<b>{{NPMSelected}}</b></h3>
        </div>
        <div class="box-body">
            <div class="box-body table-responsive no-padding">
                <table datatable="ng" class="table table-bordered table-hover">
                    <thead>
                        <tr>
                            <th>Semester</th>
                            <th>Kode</th>
                            <th>Matakuliah</th>
                            <th>SKS</th>
                            <th>Nilai Huruf</th>
                            <th>nxsks</th>
                            <th>Keterangan</th>
                        </tr>
                    </thead>

```



```

        <th>SKS</th>
        <th>Ruangan</th>
        <th>Waktu Kuliah</th>
        <th>Dosen Pengampuh</th>
    </tr>
</thead>
<tbody style="height: 200px;">
    <tr ng-repeat="item in DatasJadwal | filter: { kelas: Kelas } |
filter: CariJadwal | orderBy : 'kelas'"
    ng-click="Pilih(item)">
        <td class="{{item.ngBinding}}">{{item.smt}}</td>
        <td class="{{item.ngBinding}}">{{item.kelas}}</td>
        <td class="{{item.ngBinding}}">{{item.nmmk}}</td>
        <td class="{{item.ngBinding}}">{{item.sks}}</td>
        <td class="{{item.ngBinding}}">{{item.ruangan}}</td>
        <td
class="{{item.ngBinding}}">{{item.hari}},&nbsp;{{item.wm}}-{{item.ws}}</td>
        <td class="{{item.ngBinding}}">{{item.dsn_saji}}</td>
    </tr>
</tbody>
</table>
</div>

</div>
<div class="box-footer clearfix">
    <div class="box-header with-border">
        <button type="submit" class="btn btn-default" disabled><b>Total
SKS: {{jmsks}}</b></button>
        <div class="box-tools">
            <div class="input-group input-group-sm" style="width:
200px;">
                <input type="text" name="table_search" class="form-control
pull-right"
                placeholder="Show By Kelas" ng-model="Kelas">

                <div class="input-group-btn">
                    <button type="submit" class="btn btn-default"><i
class="fa fa-search"></i></button>
                </div>
            </div>
        </div>
    </div>
</div>
</div>
</div>

```

```

<div class="box">
  <div class="box-header with-border">
    <h3 class="box-title">Daftar Matakuliah yang Dipilih</h3>
  </div>
  <div class="box-body">
    <div class="box-body table-responsive no-padding">
      <table datatable="ng" class="table table-bordered table-hover">
        <thead>
          <tr>
            <th>Semester</th>
            <th>Matakuliah</th>
            <th>SKS</th>
            <th>Kelas</th>
            <th>Ruangan</th>
            <th>Waktu Kuliah</th>
            <th>Dosen Pengampuh</th>
          </tr>
        </thead>
        <tbody>
          <tr ng-repeat="item in Datatampung | groupBy:'smt'">
            <td>{{item.smt}}</td>
            <td>{{item.nmmk}}</td>
            <td>{{item.sks}}</td>
            <td>{{item.kelas}}</td>
            <td>{{item.ruangan}}</td>
            <td>{{item.hari}},&nbsp;{{item.wm}}-{{item.ws}}</td>
            <td>{{item.dsn_saji}}</td>
          </tr>
        </tbody>
      </table>

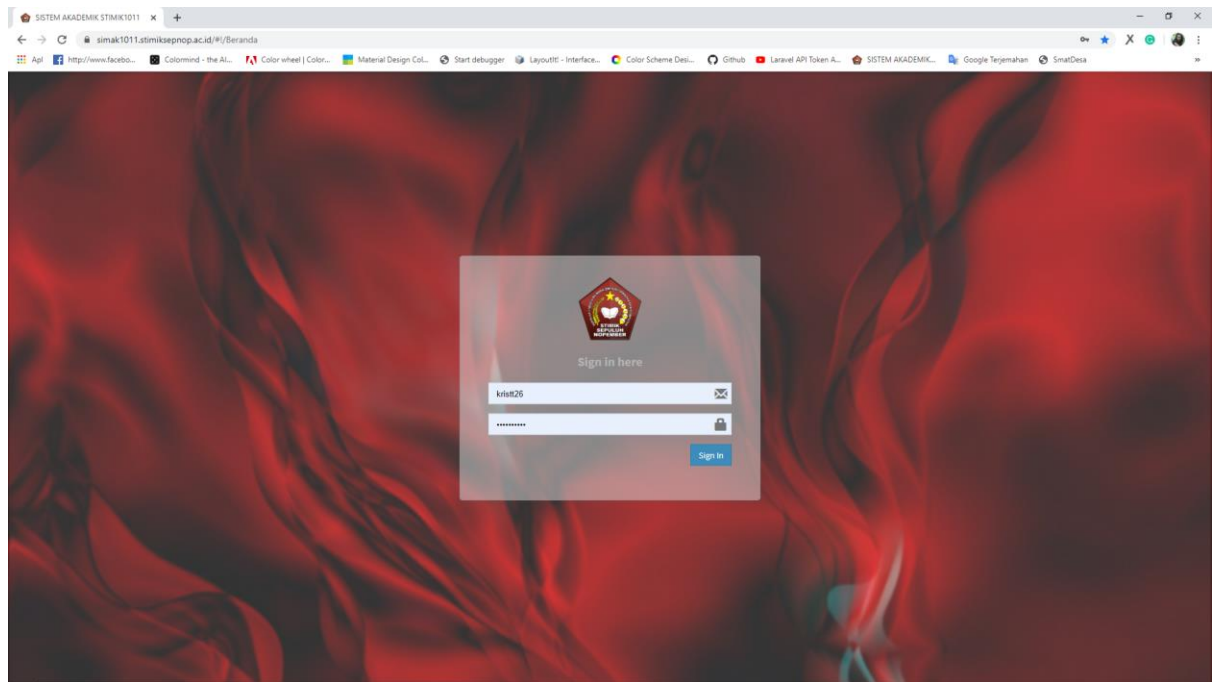
    </div>

  </div>
  <div class="box-footer clearfix">
    <button type="submit" class="btn btn-default" disabled><b>Total
    SKS: {{jmsks}}</b></button>
  </div>
</div>
</div>

```

## C. Dokumentasi

### 1. Login

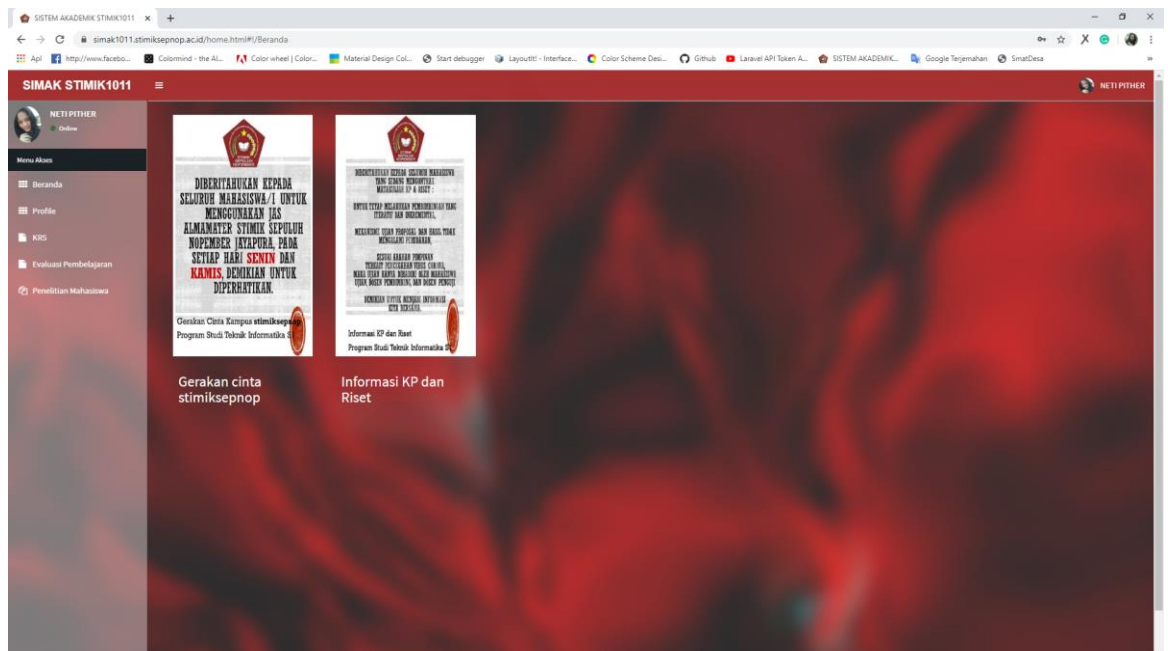


Gambar 1. Halaman Login

Halamana ini adalah halaman login yang akan digunakan oleh dosen dan mahasiswa untuk mengakses sistem. Semua dosen dan mahasiswa stimik memiliki username dan password yang di daftarkan oleh admin sistem

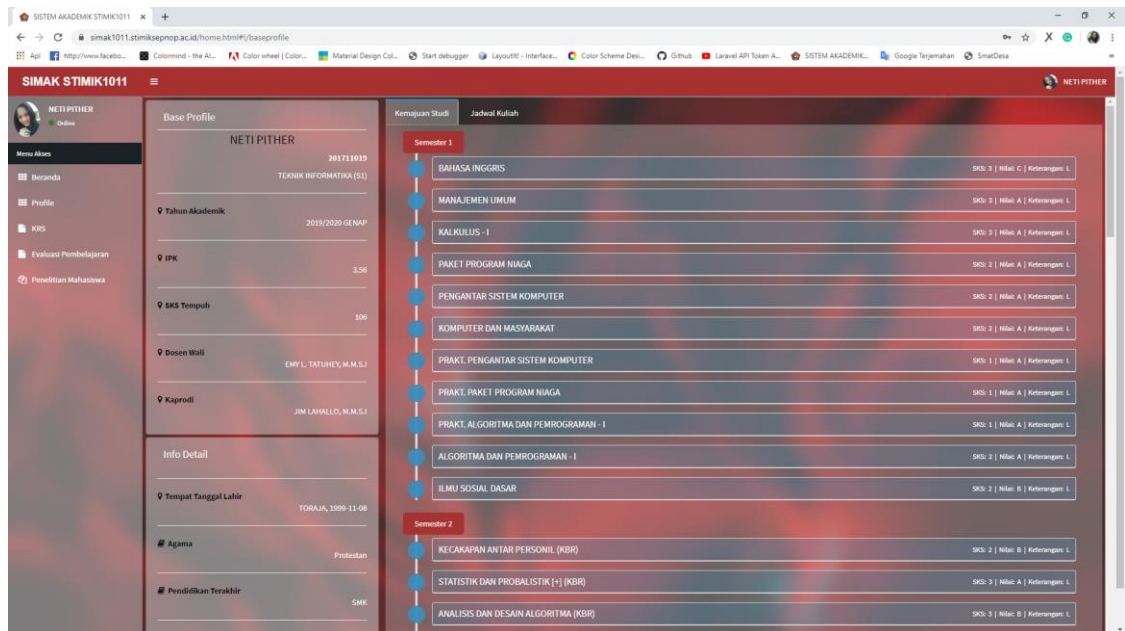
## 2. Halaman Mahasiswa

### a. Halaman Utama



Gambar 2. Halaman utama mahasiswa

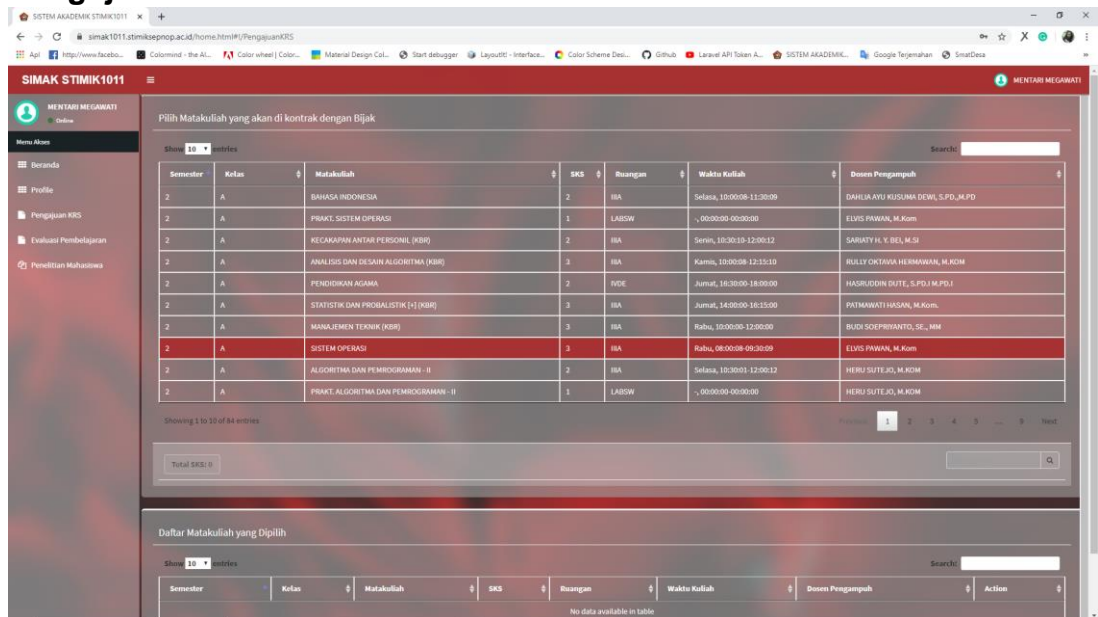
### b. Profile



Gambar 3. Halaman profile

Halaman ini berisi informasi detail tentang mahasiswa dari profile umum, jurusan, dosen wali, kemajuan studi dan jadwal mahasiswa di semester aktif

### c. Pengajuan KRS dan KRS



Gambar 4. Pengajuan KRS

Halaman ini akan muncul saat tahun akademik baru di aktifkan. Mahasiswa mengurus KRS nya melalui halaman ini, prosedur pengisian krs cukup dengan memilih matakuliah yang akan di kontrak di semester aktif kemudian ajukan untuk di validasi oleh bagian keuangan.

| No    | Mata Kuliah                                   | Kelas | Semester | SKS | Nilai | Keterangan |
|-------|---|-------|----------|-----|-------|------------|
| 1     | 000009 KEMAHIRAN                              | B     | 6        | 2   |       |            |
| 2     | 111031 JARINGAN KOMPUTER                      | B     | 6        | 3   |       |            |
| 3     | 111032 KECERDASAN BUATAN                      | B     | 6        | 2   |       |            |
| 4     | 111034 INTERAKSI MANUSIA DAN KOMPUTER         | B     | 6        | 3   |       |            |
| 5     | 111059 PERANCANGAN SISTEM INFORMASI (KIR)     | B     | 6        | 3   |       |            |
| 6     | 111060 PERANCANGAN PERANGKAT LUNAK (KIR)      | B     | 6        | 3   |       |            |
| 7     | 111231 PRAKTIK JARINGAN KOMPUTER              | B     | 6        | 1   |       |            |
| 8     | 111234 PRAKTIK INTERAKSI MANUSIA DAN KOMPUTER | B     | 6        | 1   |       |            |
| Total |   |       |          | 18  |       |            |

Gambar 5. Halaman KRS

Halaman ini akan muncul saat pengisian KRS mahasiswa telah selesai (validasi terakhir oleh bagian program studi). Halaman ini memuat daftar matakuliah yang di kontrak sesuai dengan pengisian KRS yang dilakukan.

#### d. Evaluasi Pembelajaran

Halaman ini adalah halaman yang digunakan oleh mahasiswa untuk melakukan penilaian pembelajaran dosen. Isi Halaman ini akan tampil saat periode penilaian di aktifkan oleh program studi.

#### e. Halaman penelitian

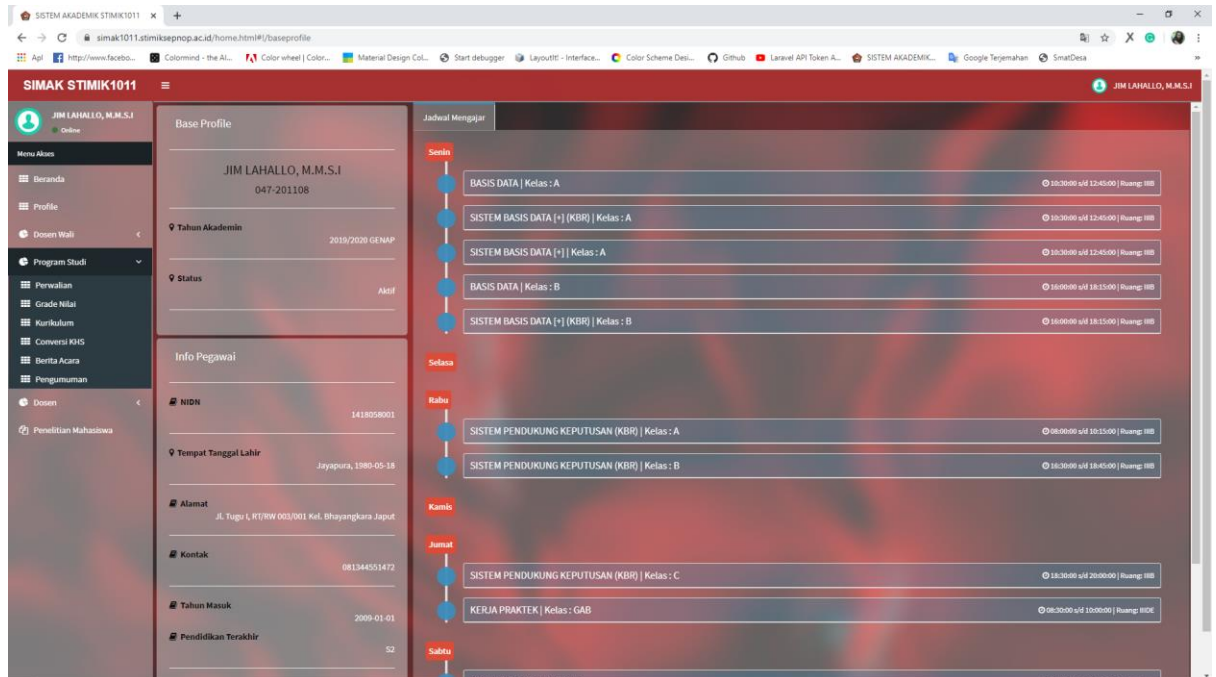
| Riset (TA)  | Karya Praktek (KP) | Publikasi |
|---|--------------------|-----------|
| <b>DAFTAR RISET MAHASISWA</b>   |                    |           |
| Sistem Pendukung Keputusan Penentuan Kelayakan Pemberian Kredit pada PT. BPR PAPUA MANDIRI MAKMUR Menggunakan Metode Weight Product<br>DIA HADIMRA, JIN LAHALLO, M.M.S.I., 2020-02-20, <a href="#">Belum ada file</a>                                     |                    |           |
| Sistem Informasi Nilai Raport Siswa SMK Hikmah Yagis Jayapura Berbasis Website<br>DIA HARINDY DAMBUS, RAHMAT HARINDY KISWANTO,ST.MT., 2020-02-21, <a href="#">Belum ada file</a>  |                    |           |
| Rancang Bangun Sistem Informasi Bengkel Waena Jaya<br>DIA MONITA SARI, HERU SUTERJO, M.KOM., 2020-02-08, <a href="#">Belum ada file</a>   |                    |           |
| Sistem Pendukung Keputusan Pemilihan Calon Ketua Skysav Menggunakan Metode Analytical Hierarchy Process (AHP)<br>DIA SIMON SAPAN SAMPELUA, RAHMAT HARINDY KISWANTO,ST.MT., 2020-01-25, <a href="#">Belum ada file</a>                                     |                    |           |
| Sistem Monitoring Pelaporan Pertanggung Jawaban Pembayaran Manfaat Program ASABRI pada PT. ASABRI (PERSERO) Cabang Jayapura<br>DIA SHAFUL HARI ACHMAD, RAHMAT HARINDY KISWANTO,ST.MT., 2020-03-20, <a href="#">Download</a>                               |                    |           |
| Sistem Pendukung Keputusan Pemilihan Calon Majelis Jemaat menggunakan Metode Profile Matching (studi kasus : Gereja GKI PNEI Kotaraja Jayapura)<br>DIA IBRAHIM RAJU KELI HEPOIN, RAHMAT HARINDY KISWANTO,ST.MT., 2019-10-02, <a href="#">Download</a>     |                    |           |
| Sistem Pendukung Keputusan Kandidat Pegawai Teladan pada Kantor Pelayanan Perbendaharaan Negara Jayapura Menggunakan Metode Analytical Hierarchy Process<br>DIA JEANY GABRIELLA SUWASE, JIN LAHALLO, M.M.S.I., 2019-07-25, <a href="#">Belum ada file</a> |                    |           |
| Rancang Bangun Aplikasi Pemetaan Ekspedisi di Jayapura Berbasis Android<br>DIA RIFKA ANUSMAN SAPUTRA, RAHMAT HARINDY KISWANTO,ST.MT., 2019-07-17, <a href="#">Download</a>  |                    |           |
| Sistem Informasi Kelayakan Angkutan Dalam Trayek di Bidang Pengujian Kendaraan Bermotor pada Kantor Dinas Perhubungan Komunikasi dan Informatika Kota Jayapura<br>DIA YANUSUS KALANABIN, WODOO, S.KOM. M.KOM., 2019-04-25, <a href="#">Belum ada file</a> |                    |           |
| Aplikasi Location Based Service Obyek Wisata Kota Jayapura Berbasis Android<br>DIA DENTY KAFAR, RAHMAT HARINDY KISWANTO,ST.MT., 2019-03-20, <a href="#">Download</a>  |                    |           |

Gambar 6. Halaman penelitin mahasiswa

Halaman ini berisi daftar tugas akhir mahasiswa dan penelitian dosen



### 3. Halaman Profile Staf / Dosen

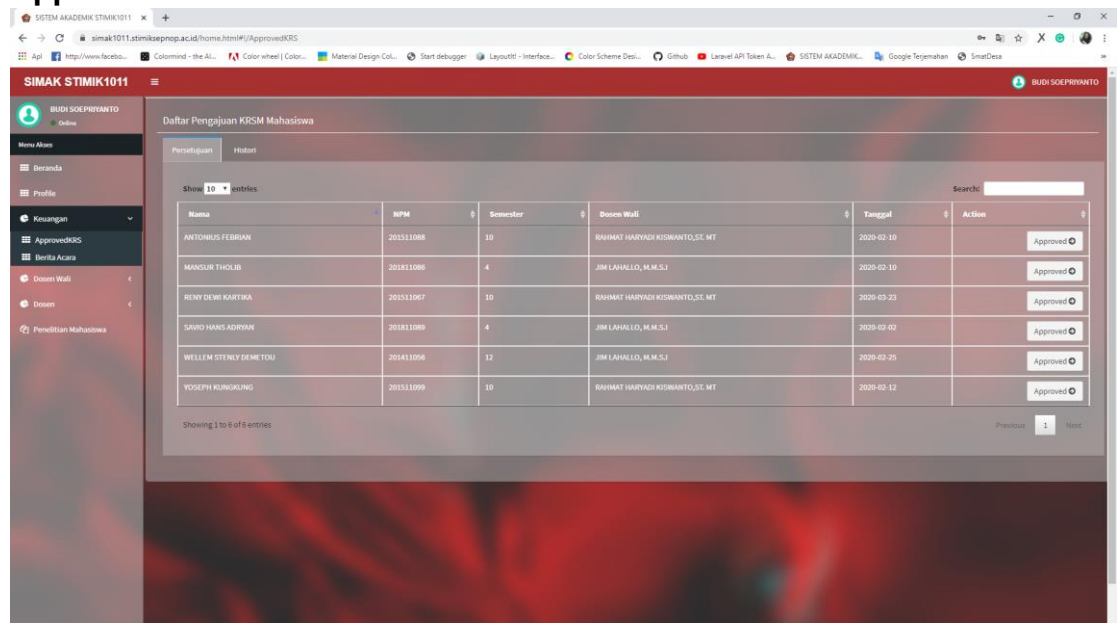


Gambar 7. Halaman Profile

Halaman ini berisi informasi mengenai staf atau dosen yang bersangkutan, khusus untuk dosen, terdapat tambahan informasi yaitu jadwal matakuliah yang di ajarkan pada semester aktif.

### 4. Halaman Bagian Keuangan

#### a. Approved KRS



Gambar 8. Halaman Approved KRS

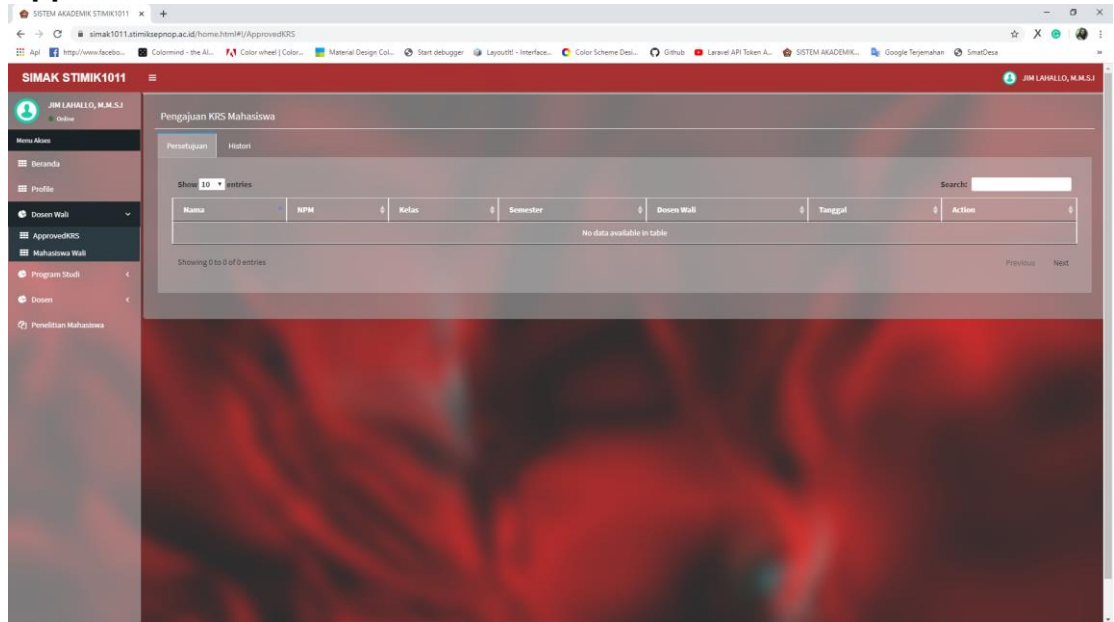
Halaman ini adalah halaman yang digunakan untuk melakukan validasi pengajuan KRS mahasiswa, prosedur validasi di dasarkan pada status pembayaran mahasiswa. Data KRS mahasiswa akan di validasi oleh bagian



keuangan saat mahasiswa telah melaksanakan kewajibannya yaitu melakukan pembayaran biaya perkuliahan.

## 5. Halaman Dosen Wali

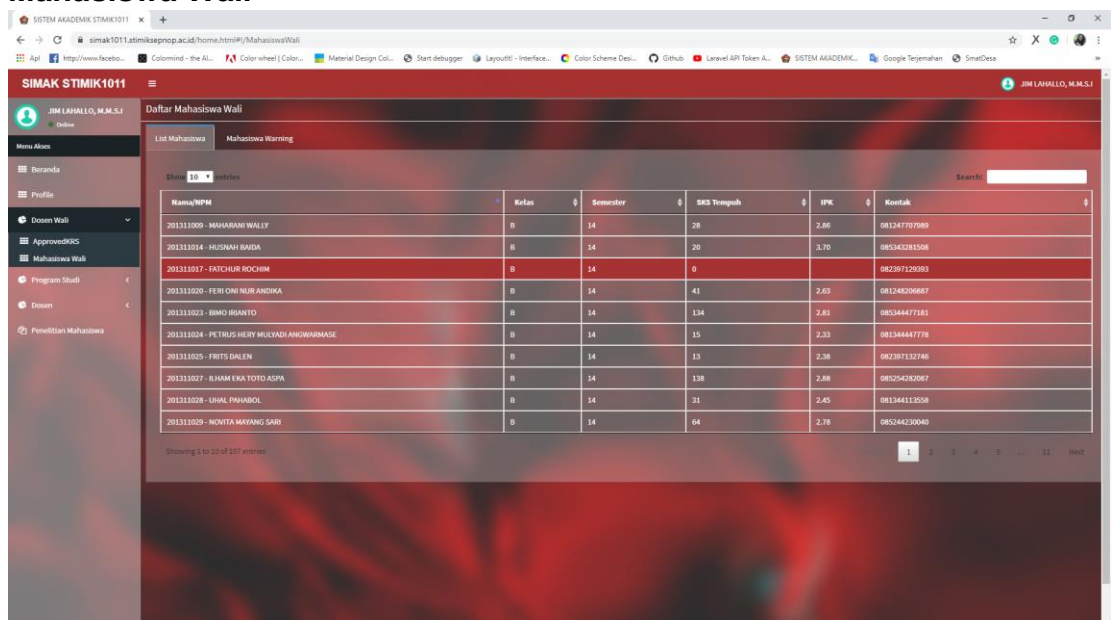
### a. Approved KRS



Gambar 9. Halaman Approved KRS

Data mahasiswa wali dosen bersangkutan akan tampil saat pengajuan KRS mahasiswa telah disetujui oleh bagian keuangan.

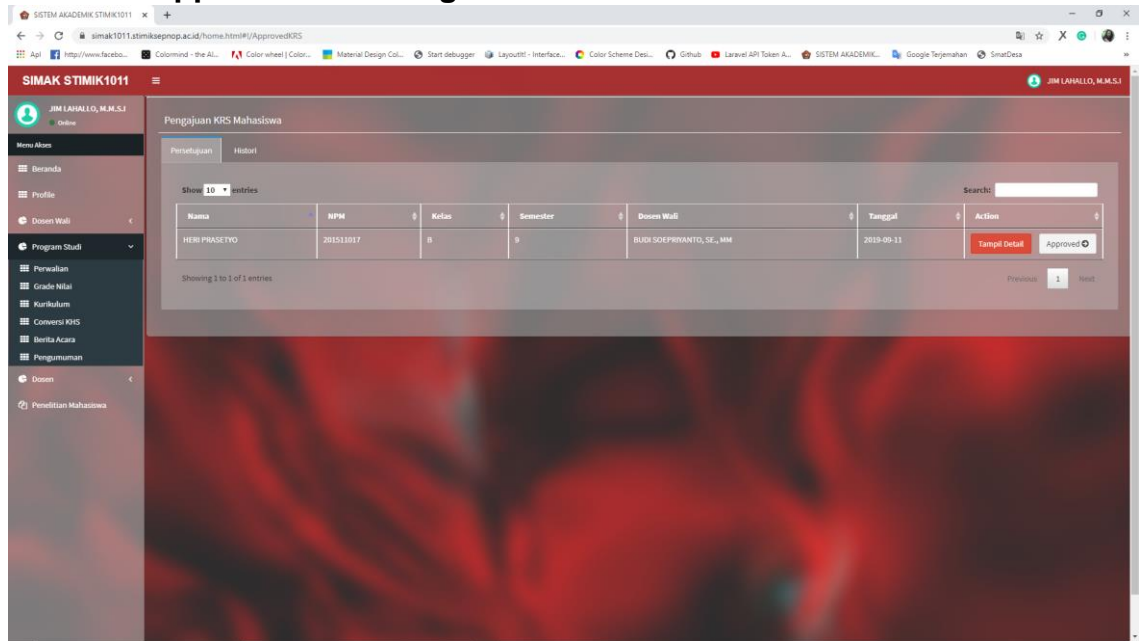
### b. Mahasiswa Wali



Gambar 10. Halaman Mahasiswa Wali

Halaman ini berisi daftar mahasiswa wali dosen bersangkutan

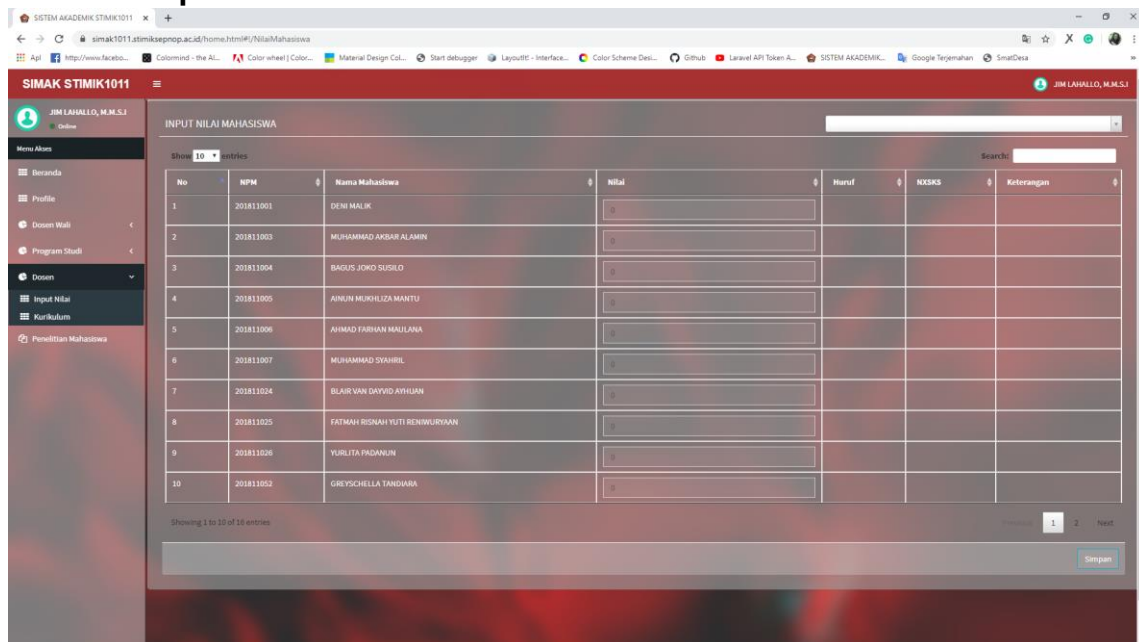
## 6. Halaman Approved KRS Program Studi



Gambar 11. Halaman Approved

Proses validasi akhir dari pengajuan KRS mahasiswa berada di bagian program studi, data mahasiswa akan tampil saat KRS Mahasiswa telah disetujui oleh Dosen wali masing-masing mahasiswa.

## 7. Halaman Input Nilai matakuliah



Gambar 12. Halaman Input nilai

Halaman ini adalah halaman yang hanya dapat diakses oleh Dosen, dimana halaman ini adalah tempat untuk mendaftarkan nilai mahasiswa berdasarkan matakuliah yang diajarkan oleh dosen yang bersangkutan.