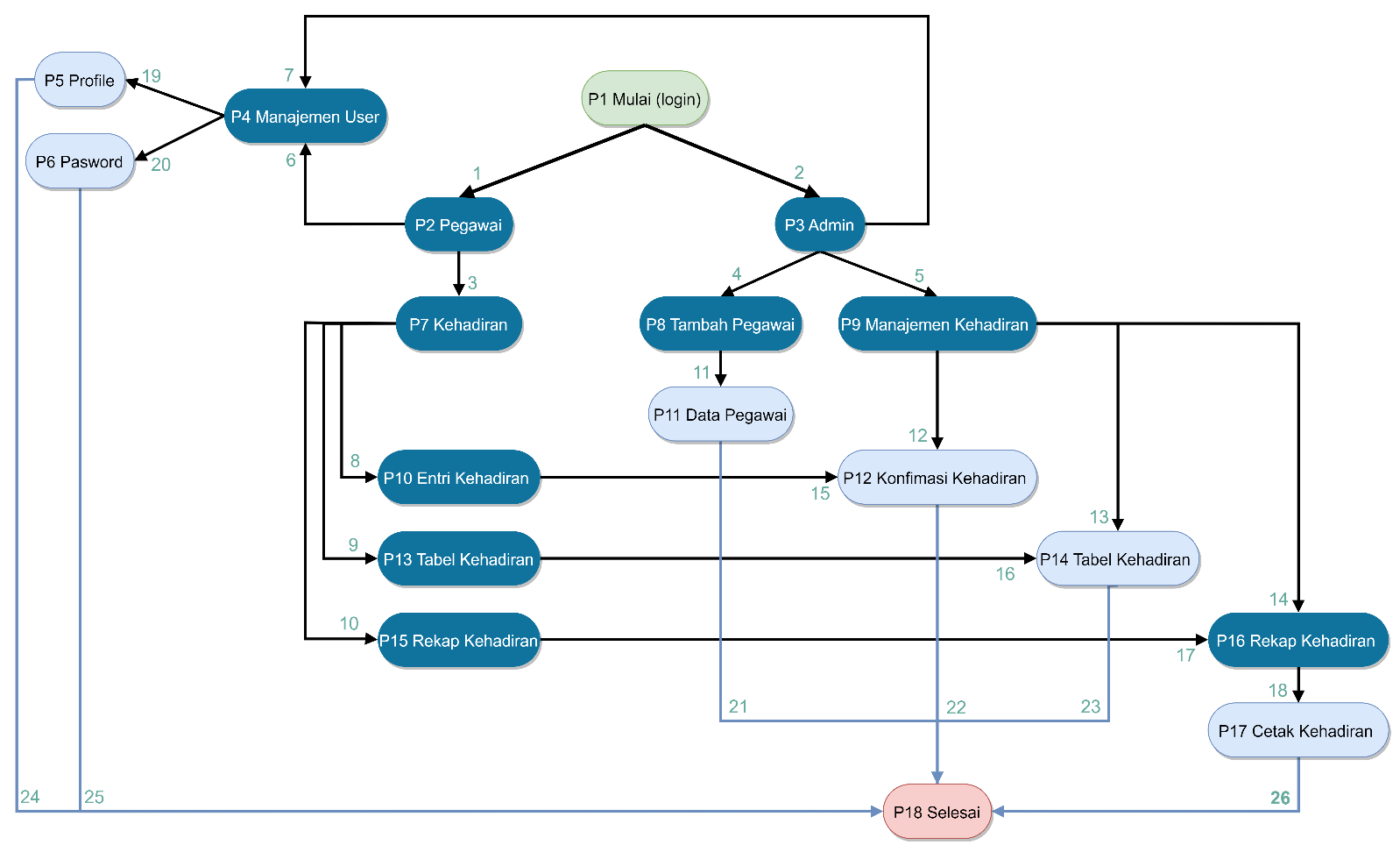
**Diagram Path**

Keseluruhan



|  |  |
| --- | --- |
| Login | Absensi |
|  |  |

**Listing Source**

Login

if($this->form\_validation->run() == FALSE){

            $this->load->view('auth/login');

        }else{

            $password = $this->input->post('password');

            $user = $this->auth->checkEmail();

*// Cek user berdasarkan email*

            if(isset($user)){

*// Jika ada*

*// Cek password sesuai atau tidak*

                if(hashEncryptVerify($password, $user['password']) == TRUE){

                    $this->session->set\_userdata($user);

                    $this->session->set\_userdata('login', TRUE);

                    if($user['role\_id'] == 1){

                        redirect('admin');

                  }else{

                        redirect('user');

                  }

                }else{

*// Jika password tidak sesuai*

                    $this->session->set\_flashdata('message',

                    '<div class="alert alert-danger alert-dismissible fade show" role="alert">

                        Email atau password salah!

                        <button type="button" class="close" data-dismiss="alert" aria-label="Close">

                            <span aria-hidden="true">&times;</span>

                        </button>

                    </div>');

                    redirect('auth');

                }

            }else{

*// Jika email tidak sesuai*

                $this->session->set\_flashdata('message',

                    '<div class="alert alert-danger alert-dismissible fade show" role="alert">

                        Email atau password salah!

                        <button type="button" class="close" data-dismiss="alert" aria-label="Close">

                            <span aria-hidden="true">&times;</span>

                        </button>

                    </div>');

                redirect('auth');

            }

        }

    }

Pegawai

public function profile()

    {

        $this->form\_validation->set\_rules('name', 'Nama', 'required|trim',[

            'required' => 'Nama tidak boleh kosong.'

        ]);

        $this->form\_validation->set\_rules('email', 'Email', 'required|trim|valid\_email',[

            'required' => 'Email tidak boleh kosong.'

        ]);

        if($this->form\_validation->run() == FALSE){

            $id = $this->session->userdata('id\_users');

            $user = $this->profile->getProfile($id);

            $data['title']      = 'Biodata';

            $data['page']       = 'user/profile/profile';

            $data['user']       = $user;

            $data['position'] = $this->anggota->getPosition();

            $this->load->view('templates/app', $data);

        }else{

            $id = $this->input->post('id');

            $data = [

                'name'  => $this->input->post('name'),

                'email'     => $this->input->post('email'),

                'gender'    => $this->input->post('gender'),

            ];

*// Jika foto diubah*

            if(!empty($\_FILES['photo']['name'])){

                $upload      = $this->profile->uploadImage();

*// Jika upload berhasil*

                if($upload){

*// Ambil data user*

                    $imageProfile = $this->profile->getProfile($id);

*// Hapus foto user sebelum di update*

                    if(file\_exists('image/' . $imageProfile['photo']) && $imageProfile['photo']){

                        unlink('image/' . $imageProfile['photo']);

                    }

*// Timpa data foto dengan nama yg baru*

                    $data['photo'] = $upload;

                }else{

                    redirect(base\_url("user/profile"));

                }

            }

            $this->profile->updateProfile($id, $data);

            $this->session->set\_flashdata('message', 'Biodata Berhasil Diupdate. Silahkan login kembali untuk memperbaharui profile.');

            redirect(base\_url('user/profile'));

        }

    }

    public function change\_password()

    {

        $this->form\_validation->set\_rules('new\_password', 'Password Baru', 'required|trim',[

            'required' => 'Password baru tidak boleh kosong.',

        ]);

        $this->form\_validation->set\_rules('password\_confirm', 'Konfirmasi Password', 'required|trim|matches[new\_password]',[

            'required' => 'Konfirmasi password tidak boleh kosong.',

            'matches'  => 'Konfirmasi password tidak sesuai'

        ]);

        if($this->form\_validation->run() == FALSE){

            $data['title']      = 'Ubah Password';

            $data['page']       = 'user/profile/password';

            $this->load->view('templates/app', $data);

        }else{

            $id = $this->session->userdata('id\_users');

            $data = [

                'password' => hashEncrypt($this->input->post('new\_password')),

            ];

            $this->user->updatePassword($id, $data);

            $this->session->set\_flashdata('message', 'Password berhasil diupdate.');

            redirect(base\_url('user'));

        }

    }

Pegawai: Kehadiran

public function entri()

    {

        $data['title'] = 'Entri Kehadiran';

        $data['page']   = 'user/kehadiran/entri';

        $this->load->view('templates/app', $data);

    }

    public function masuk()

    {

        $data = [

            'user\_id'       => $this->session->userdata('id\_users'),

            'date'          => date('Y-m-d'),

            'time'          => date('H:i:s'),

            'information'   => 'M',

            'status'            => 0

        ];

        $this->absensi->insertEntri($data);

        $this->session->set\_flashdata('message', 'Entri kehadiran berhasil. Silahkan tunggu konfirmasi oleh administator.');

        redirect(base\_url('absensi/entri'));

    }

    public function ijin()

    {

        $data = [

            'user\_id'       => $this->session->userdata('id\_users'),

            'date'          => date('Y-m-d'),

            'time'          => date('H:i:s'),

            'information'   => 'I',

            'status'            => 0

        ];

        $this->absensi->insertEntri($data);

        $this->session->set\_flashdata('message', 'Entri kehadiran berhasil. Silahkan tunggu konfirmasi oleh administator.');

        redirect(base\_url('absensi/entri'));

    }

    public function sakit()

    {

        $data = [

            'user\_id'       => $this->session->userdata('id\_users'),

            'date'          => date('Y-m-d'),

            'time'          => date('H:i:s'),

            'information'   => 'S',

            'status'            => 0

        ];

        $this->absensi->insertEntri($data);

        $this->session->set\_flashdata('message', 'Entri kehadiran berhasil. Silahkan tunggu konfirmasi oleh administator.');

        redirect(base\_url('absensi/entri'));

    }

    public function tabel()

    {

        $data['title']  = 'Tabel Kehadiran';

        $data['page']       = 'user/kehadiran/tabel';

        $data['absensi']    = $this->absensi->getAbsensi();

        $this->load->view('templates/app', $data);

    }

    public function rekap()

    {

        $data['title']  = 'Rekap Kehadiran';

        $data['page']       = 'user/kehadiran/rekap';

        $data['hadir']  = $this->absensi->getHadir();

        $data['ijin']       = $this->absensi->getIjin();

        $data['sakit']  = $this->absensi->getSakit();

        $this->load->view('templates/app', $data);

    }

Admin: Manajemen User

public function profile()

    {

        $this->form\_validation->set\_rules('email', 'Email', 'required|trim|valid\_email|is\_unique[users.email]',[

            'required' => 'Email tidak boleh kosong.',

            'is\_unique'=> 'Email sudah terdaftar.'

        ]);

        if($this->form\_validation->run() == FALSE){

            $data['title']      = 'Biodata';

            $data['page']       = 'admin/profile/profile';

            $data['admin']      = $this->admin->getAdminProfile();

            $data['position'] = $this->admin->getPosition();

            $this->load->view('templates/app', $data);

        }else{

            $data = [

                'email' => $this->input->post('email'),

            ];

            $this->admin->updateProfile($data);

            $this->session->set\_flashdata('message', 'Data admin berhasil diupdate.');

            redirect(base\_url('admin'));

        }

    }

    public function change\_password()

    {

        $this->form\_validation->set\_rules('new\_password', 'Password Baru', 'required|trim',[

            'required' => 'Password baru tidak boleh kosong.',

        ]);

        $this->form\_validation->set\_rules('password\_confirm', 'Konfirmasi Password', 'required|trim|matches[new\_password]',[

            'required' => 'Konfirmasi password tidak boleh kosong.',

            'matches'  => 'Konfirmasi password tidak sesuai'

        ]);

        if($this->form\_validation->run() == FALSE){

            $data['title']      = 'Ubah Password';

            $data['page']       = 'admin/profile/password';

            $this->load->view('templates/app', $data);

        }else{

            $data = [

                'password' => hashEncrypt($this->input->post('new\_password')),

            ];

            $this->admin->updatePassword($data);

            $this->session->set\_flashdata('message', 'Password berhasil diupdate.');

            redirect(base\_url('admin'));

        }

    }

Admin: Tambah Pegawai

public function dosen() {

        $data['title'] = 'Dosen';

        $data['page']  = 'admin/anggota/index';

        $data['users'] = $this->anggota->getDosen();

        $this->load->view('templates/app', $data);

    }

    public function tatausaha() {

        $data['title'] = 'TataUsaha';

        $data['page']  = 'admin/anggota/index';

        $data['users'] = $this->anggota->getTataUsaha();

        $this->load->view('templates/app', $data);

    }

    public function staff() {

        $data['title'] = 'Staff';

        $data['page']  = 'admin/anggota/index';

        $data['users'] = $this->anggota->getStaff();

        $this->load->view('templates/app', $data);

    }

    public function laboran() {

        $data['title'] = 'Laboran';

        $data['page']  = 'admin/anggota/index';

        $data['users'] = $this->anggota->getLaboran();

        $this->load->view('templates/app', $data);

    }

    public function add() {

        $this->form\_validation->set\_rules('no\_employee', 'No Karyawan', 'required|trim|is\_unique[users.no\_employee]', [

            'required'  => 'No karyawan tidak boleh kosong.',

            'is\_unique' => 'No karyawan sudah terdaftar.',

        ]);

        $this->form\_validation->set\_rules('name', 'Nama', 'required|trim', [

            'required' => 'Nama tidak boleh kosong.',

        ]);

        $this->form\_validation->set\_rules('email', 'Email', 'required|trim|valid\_email|is\_unique[users.email]', [

            'required'    => 'Email tidak boleh kosong.',

            'valid\_email' => 'Email tidak valid',

            'is\_unique'   => 'Email sudah terdaftar.',

        ]);

        $this->form\_validation->set\_rules('password', 'Password', 'required',[

            'required'      => 'Password tidak boleh kosong'

        ]);

        if ($this->form\_validation->run() == FALSE) {

            $data['title']    = 'Tambah Anggota';

            $data['page']     = 'admin/anggota/add';

            $data['position'] = $this->anggota->getPosition();

            $this->load->view('templates/app', $data);

        } else {

            $data = [

                'no\_employee'  => $this->input->post('no\_employee'),

                'name'         => $this->input->post('name'),

                'gender'       => $this->input->post('gender'),

                'email'        => strtolower($this->input->post('email')),

                'photo'        => null,

                'password'     => hashEncrypt($this->input->post('password')),

                'role\_id'      => 2,

                'position\_id'  => $this->input->post('position\_id'),

                'date\_created' => date('Y-m-d'),

            ];

            $this->anggota->insertUser($data);

            $this->session->set\_flashdata('message', 'Data anggota berhasil ditambahkan.');

            if ($data['position\_id'] == 1) {

                redirect(base\_url('anggota/dosen'));

            } else if ($data['position\_id'] == 2) {

                redirect(base\_url('anggota/tatausaha'));

            } else if ($data['position\_id'] == 3) {

                redirect(base\_url('anggota/staff'));

            } else if ($data['position\_id'] == 4) {

                redirect(base\_url('anggota/laboran'));

            }

        }

    }

    public function edit($id) {

        $this->form\_validation->set\_rules('no\_employee', 'No Karyawan', 'required|trim', [

            'required' => 'No karyawan tidak boleh kosong.',

        ]);

        $this->form\_validation->set\_rules('name', 'Nama', 'required|trim', [

            'required' => 'Nama tidak boleh kosong.',

        ]);

        if ($this->form\_validation->run() == FALSE) {

            $data['title']    = 'Ubah Anggota';

            $data['page']     = 'admin/anggota/edit';

            $data['user']     = $this->anggota->getDetailUser($id);

            $data['position'] = $this->anggota->getPosition();

            $this->load->view('templates/app', $data);

        } else {

            $data = [

                'no\_employee' => $this->input->post('no\_employee'),

                'name'        => $this->input->post('name'),

                'gender'      => $this->input->post('gender'),

                'position\_id' => $this->input->post('position\_id'),

            ];

            $this->anggota->updateUser($id, $data);

            $this->session->set\_flashdata('message', 'Data anggota berhasil ditambahkan.');

            if ($data['position\_id'] == 1) {

                redirect(base\_url('anggota/dosen'));

            } else if ($data['position\_id'] == 2) {

                redirect(base\_url('anggota/tatausaha'));

            } else if ($data['position\_id'] == 3) {

                redirect(base\_url('anggota/staff'));

            } else if ($data['position\_id'] == 4) {

                redirect(base\_url('anggota/laboran'));

            }

        }

    }

    public function delete($id) {

        $user = $this->anggota->getDetailUser($id);

        $this->anggota->deleteUser($id);

        $this->session->set\_flashdata('message', 'Data anggota berhasil dihapus.');

        if ($user['position\_id'] == 1) {

            redirect(base\_url('anggota/dosen'));

        } else if ($user['position\_id'] == 2) {

            redirect(base\_url('anggota/tatausaha'));

        } else if ($user['position\_id'] == 3) {

            redirect(base\_url('anggota/staff'));

        } else if ($user['position\_id'] == 4) {

            redirect(base\_url('anggota/laboran'));

        }

    }

}

Admin: Manajemen Kehadiran

public function konfirmasi\_kehadiran() {

        $data['title']   = 'Konfirmasi Kehadiran';

        $data['page']    = 'admin/kehadiran/konfirmasi';

        $data['absensi'] = $this->kehadiran->getKehadiran();

        $this->load->view('templates/app', $data);

    }

    public function konfirmasi($id) {

        $data           = $this->kehadiran->getAbsensi($id);

        $data['status'] = 1;

        $this->kehadiran->updateAbsensi($id, $data);

        $this->session->set\_flashdata('message', 'Absensi berhasil dikonfirmasi.');

        redirect(base\_url('kehadiran/konfirmasi\_kehadiran'));

    }

    public function tolak($id) {

        $data           = $this->kehadiran->getAbsensi($id);

        $data['status'] = 2;

        $this->kehadiran->updateAbsensi($id, $data);

        $this->session->set\_flashdata('message', 'Absensi berhasil ditolak.');

        redirect(base\_url('kehadiran/konfirmasi\_kehadiran'));

    }

    public function tabel\_kehadiran() {

        $data['title']   = 'Tabel Kehadiran';

        $data['page']    = 'admin/kehadiran/tabel';

        $data['absensi'] = $this->kehadiran->getAllKehadiranPerDay();

        $this->load->view('templates/app', $data);

    }

    public function rekap\_kehadiran() {

        $data['title'] = 'Rekap Kehadiran';

        $data['page']  = 'admin/kehadiran/rekap';

        $data['rekap'] = $this->kehadiran->getRekap();

        $this->load->view('templates/app', $data);

    }

Admin: Rekap Kehadiran (Cetak)

public function cetak() {

*// make a new spreadsheet object*

        $spreadsheet = new Spreadsheet();

*//get current active sheet (first sheet)*

        $sheet = $spreadsheet->getActiveSheet();

*//set default font*

        $spreadsheet->getDefaultStyle()

            ->getFont()

            ->setName('Arial')

            ->setSize(10)

        ;

*//heading*

        $spreadsheet->getActiveSheet()

            ->setCellValue('A1', "Rekap Absensi Program Studi");

*//merge heading*

        $spreadsheet->getActiveSheet()->mergeCells("A1:G1");

*// set font style*

        $spreadsheet->getActiveSheet()->getStyle('A1')->getFont()->setSize(20);

*// set cell alignment*

        $spreadsheet->getActiveSheet()->getStyle('A1')->getAlignment()->setHorizontal(Alignment::HORIZONTAL\_CENTER);

*//setting column width*

        $spreadsheet->getActiveSheet()->getColumnDimension('A')->setWidth(10);

        $spreadsheet->getActiveSheet()->getColumnDimension('B')->setWidth(20);

        $spreadsheet->getActiveSheet()->getColumnDimension('C')->setWidth(20);

        $spreadsheet->getActiveSheet()->getColumnDimension('D')->setWidth(10);

        $spreadsheet->getActiveSheet()->getColumnDimension('E')->setWidth(10);

        $spreadsheet->getActiveSheet()->getColumnDimension('F')->setWidth(10);

        $spreadsheet->getActiveSheet()->getColumnDimension('G')->setWidth(10);

*//header text*

        $spreadsheet->getActiveSheet()

            ->setCellValue('A2', "No")

            ->setCellValue('B2', "Nama")

            ->setCellValue('C2', "Bagian")

            ->setCellValue('D2', "Hadir")

            ->setCellValue('E2', "Sakit")

            ->setCellValue('F2', "Ijin")

            ->setCellValue('G2', "Total");

        $data = $this->kehadiran->getRekap();

*// echo '<pre>' . var\_export($data, true) . '</pre>'; die;*

*//loop through the data*

        $row    = 3; *//current row*

        $no     = 1;

        $indeks = 0;

        foreach ($data as $d) {

            $spreadsheet->getActiveSheet()

                ->setCellValue('A' . $row, $no++)

                ->setCellValue('B' . $row, $data[$indeks]['name'])

                ->setCellValue('C' . $row, $data[$indeks]['position'])

                ->setCellValue('D' . $row, $data[$indeks]['M'])

                ->setCellValue('E' . $row, $data[$indeks]['S'])

                ->setCellValue('F' . $row, $data[$indeks]['I'])

                ->setCellValue('G' . $row, $data[$indeks]['total'])

            ;

            $row++;

            $indeks++;

        }

        header('Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');

        header('Content-Disposition: attachment;filename="Rekap Absensi.xlsx"');

        $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');

        $writer->save('php://output');

        exit;

    }